1. Report No. FHWA/TX-03/4020-1	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle SOFTWARE FOR TIMING SIGNALIZED ARTERIALS		5. Report Date September 2002
		6. Performing Organization Code
7. Author(s) Nadeem A. Chaudhary, Vijay G. Kovvali, Chi-Leung Chu, Joonhyo Kim, and S. M. Mahabubul Alam		8. Performing Organization Report No. Report 4020-1
9. Performing Organization Name and Address Texas Transportation Institute The Texas A&M University System College Station, Texas 77843-3135		10. Work Unit No. (TRAIS)
		11. Contract or Grant No. Project No. 0-4020
12. Sponsoring Agency Name and Address Texas Department of Transportation Research and Technology Implementation Office P. O. Box 5080 Austin, Texas 78763-5080		13. Type of Report and Period Covered Research: September 2000 – August 2001
		14. Sponsoring Agency Code

15. Supplementary Notes
Research performed in cooperation with the Texas Department of Transportation and the U.S. Department of Transportation, Federal Highway Administration.
Research Project Title: Improvements to Signal Timing Software

16. Abstract
This report documents the results of research conducted in a two-year TxDOT project dealing with the coordination of traffic signals on signalized arterials. This TxDOT project had two primary research objectives. The first objective was to develop guidelines for selecting software for use in signal timing projects. In order to achieve this objective, the researchers compared three popular signal timing optimization programs. These programs are Synchro, TRANSYT 7F, and PASSER II. The comparison was performed using several real arterials. The second objective of the project was to develop an enhanced version of PASSER II. This objective was achieved by developing a new graphic user interface, by re-engineering the underlying technology used in PASSER II-90 for maximizing arterial progression, and by developing a new delay estimation routine for undersaturated and oversaturated arterials. The new program has been named PASSER V-03. In addition to the above features, PASSER V provides several new models. These include: a genetic-algorithm-based optimizer for timing signals for optimum delay or progression, a re-engineered version of PASSER III technology for timing diamond interchanges using a single controller, and a tool for locating potential bottlenecks and for determining maximum throughput capacity of network.

17. Key Words Signalized Arterial, Delay Minimization, Progression, Bandwidth Maximization, Traffic Signal Coordination	18. Distribution Statement No restrictions. This document is available to the public through NTIS: National Technical Information Service 5285 Port Royal Road Springfield, Virginia 22161		
19. Security Classif.(of this report) Unclassified	20. Security Classif.(of this page) Unclassified	21. No. of Pages 120	22. Price

Form DOT F 1700.7 (8-72)   Reproduction of completed page authorized

SOFTWARE FOR TIMING SIGNALIZED ARTERIALS

by

Nadeem A. Chaudhary, P.E.
Research Engineer
Texas Transportation Institute

Vijay G. Kovvali
Graduate Research Assistant
Texas Transportation Institute

Chi-Leung Chu
Graduate Research Assistant
Texas Transportation Institute

Joonhyo Kim
Graduate Research Assistant
Texas Transportation Institute

and

S. M. Mahabubul Alam
Graduate Research Assistant
University of Texas at El Paso

DISCLAIMER

The contents of this report reflect the views of the authors, who are solely responsible for the facts and accuracy of the data, the opinions, and the conclusions presented herein. The contents do not necessarily reflect the official view or policies of the Texas Department of Transportation (TxDOT), the Federal Highway Administration (FHWA), the Texas A&M University System, the Texas Transportation Institute (TTI), the University of Texas System, or the University of Texas at El Paso (UTEP). This report does not constitute a standard or regulation, and its contents are not intended for construction, bidding, or permit purposes. The use and names of specific products or manufacturers listed herein does not imply endorsement of those products or manufacturers. The engineer in charge of the project was Nadeem A. Chaudhary, P.E. # 66470.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

BACKGROUND

Coordinating two or more signals on a signalized arterial requires the determination of the following four signal timing parameters to achieve the desired results or objectives:

- cycle length,
- green splits,
- phase sequence or order, and
- offsets.

The desired objectives of coordination may include one or more of the following objectives:

- providing/maintaining safety,
- minimizing delay,
- maximizing progression efficiency,
- minimizing queue size at approaches, and
- maximizing system throughput.

Key Considerations in Selecting Signal Timing Parameters

Cycle length is the most important factor for timing traffic signals. Maintaining stable flow of traffic from one signal through an adjacent signal on an arterial implicitly requires that all signals in a coordinated system operate under a common cycle length. As a result, some restrictions have to be placed on cycle lengths of individual signals. These restrictions can be established using Webster's theory for isolated traffic signals (1). According to this theory, the absolute minimum (critical) cycle length of an undersaturated traffic signal can be calculated as follows:

$$C_c = \frac{L}{1-Y}$$
Equation 1

where:
C_c = Critical cycle length,
L = Total lost time for all critical phases, and
Y = Sum of flow (volume/saturation flow) ratios for all critical phases.

According to Webster et al. (1), minimum delay cycle length for a pretimed signal can be calculated using the following formula:

$$C_m = \frac{1.5L - 5}{1 - Y}$$
Equation 2

The reader should note that C_m is approximately 50 percent higher than C_c. A variation of Equation 2 for actuated signal is (2):

1

$$C_m = \frac{1.3L - 5}{1 - Y} \qquad \text{Equation 3}$$

Equation 3 accounts for shortening of cycle length in the field due to phase gap-out and phase skipping. In actuated-coordinated signals, the cycle length remains unchanged because any slack time gets allocated to the coordinated phases.

Figure 1 illustrates a hypothetical delay versus cycle length curve for an isolated signal (*1*). This figure also identifies critical and minimum-delay cycle lengths. The location of this curve depends on traffic volumes. For instance, higher traffic volumes will shift this curve to the right and up. However, the following general characteristics apply to all situations:

- delay increases sharply for decreases in cycle lengths below C_c,
- the curve is flat in the vicinity of C_m, and
- delay curve on the right of C_m has a lower slope than that on the left of C_m.
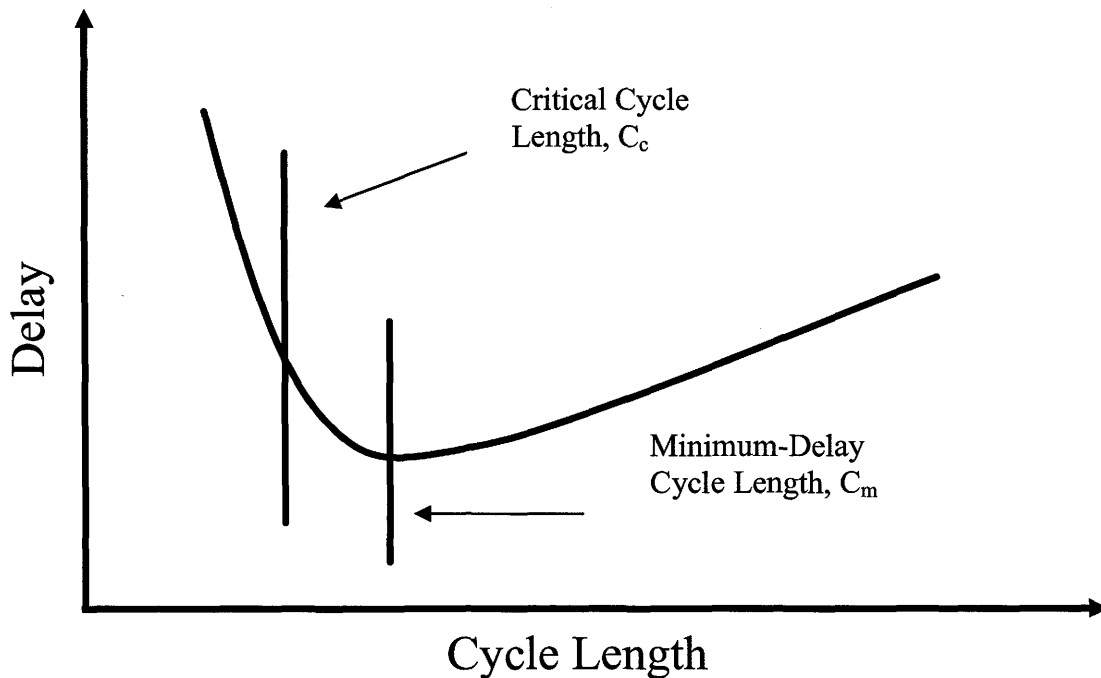
Figure 1. Shape of a Typical Delay versus Cycle Length Curve for an Isolated Signal.

In a system of signals under consideration for coordination, minimum delay cycle length for each signal will be different from other signals. This difference will require selection of a cycle length that compromises the operation of individual signals but is needed to provide good coordination that will benefit all signals. The selection of a common cycle length should ensure that no signal in the system causes excessive delays to motorists. We illustrate the initial process

of cycle length selection for one possible scenario using Figure 2. This figure shows hypothetical cycle length versus delay curves for three signals with different minimum delay cycle lengths. Since each curve is flat in the vicinity of the minimum delay cycle length, some deviation from the minimum delay cycle length will not cause a significant increase in delay. In general, any variation within 85 percent and 120 percent of the minimum delay cycle length will not adversely affect delay. As pointed out earlier, the rate of increase to the left of the minimum delay point is higher than that on the right side. This difference is because cycle lengths smaller than some minimum (generally smaller than 70 percent of the minimum) do not provide sufficient capacity to handle the demand. For this reason, larger cycle lengths can be selected, if needed. One generally accepted criterion is to select the largest minimum-delay (*Maximin*) cycle length as the system cycle length. However, this cycle length may not provide good coordination because of constraints placed by link travel times and other signal timing parameters. Therefore, it is desirable to establish a cycle-length range based on the *Maximin* cycle length. Figure 2 illustrates the minimum acceptable system cycle length, which is equal to the minimum acceptable cycle length for the signal with *Maximin* cycle. In selecting the upper limit on cycle length range, the analyst should remember that larger cycle lengths result in larger red times for signal phases, resulting in increased delay per cycle, and larger cycle-by-cycle queues. Thus, unnecessarily large cycle lengths should be avoided, especially when a system contains short links. As a general rule, the analyst should use caution when using cycle lengths of larger than 120 seconds.
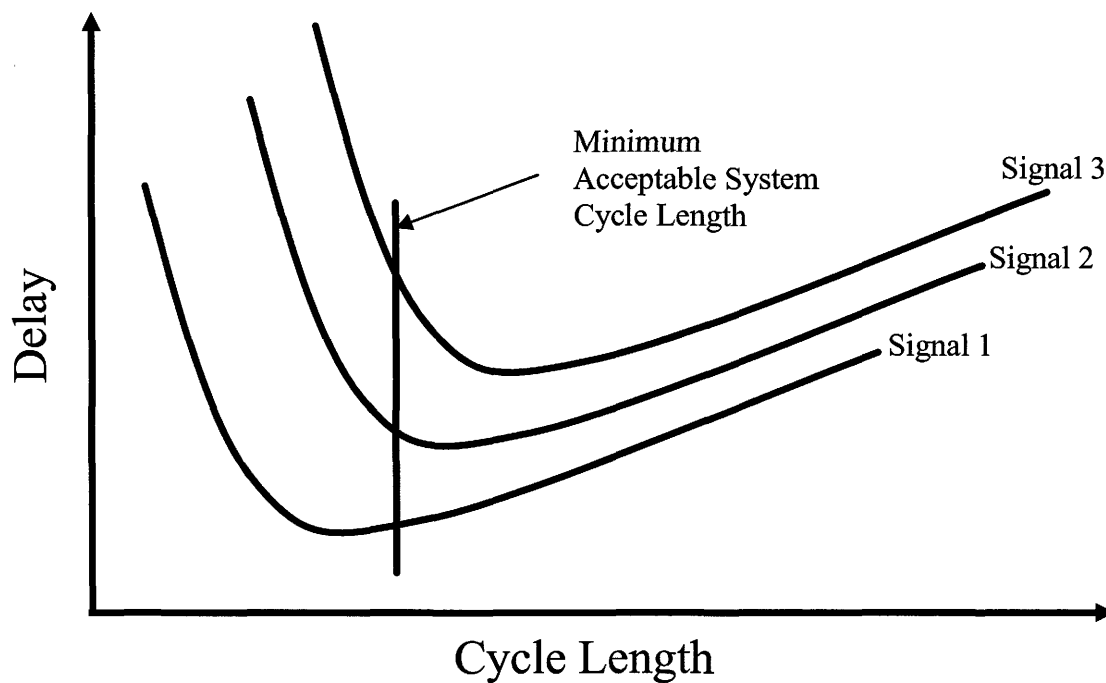
Figure 2. Constraints on System Cycle Length.

Figure 3 illustrates another common scenario where the minimum delay cycle length for at least one signal (Signal 1) in the system is significantly different from those for other signals. In such cases, the analyst can divide the system or coordinate using half/double cycling. One can

implement half/double cycling by operating a group of signals having similar characteristics using a common cycle length and each of the remaining signals using a cycle length that is a half or double of the common cycle length. Link length and traffic volumes play a major role in deciding whether to coordinate such a signal with other signals.
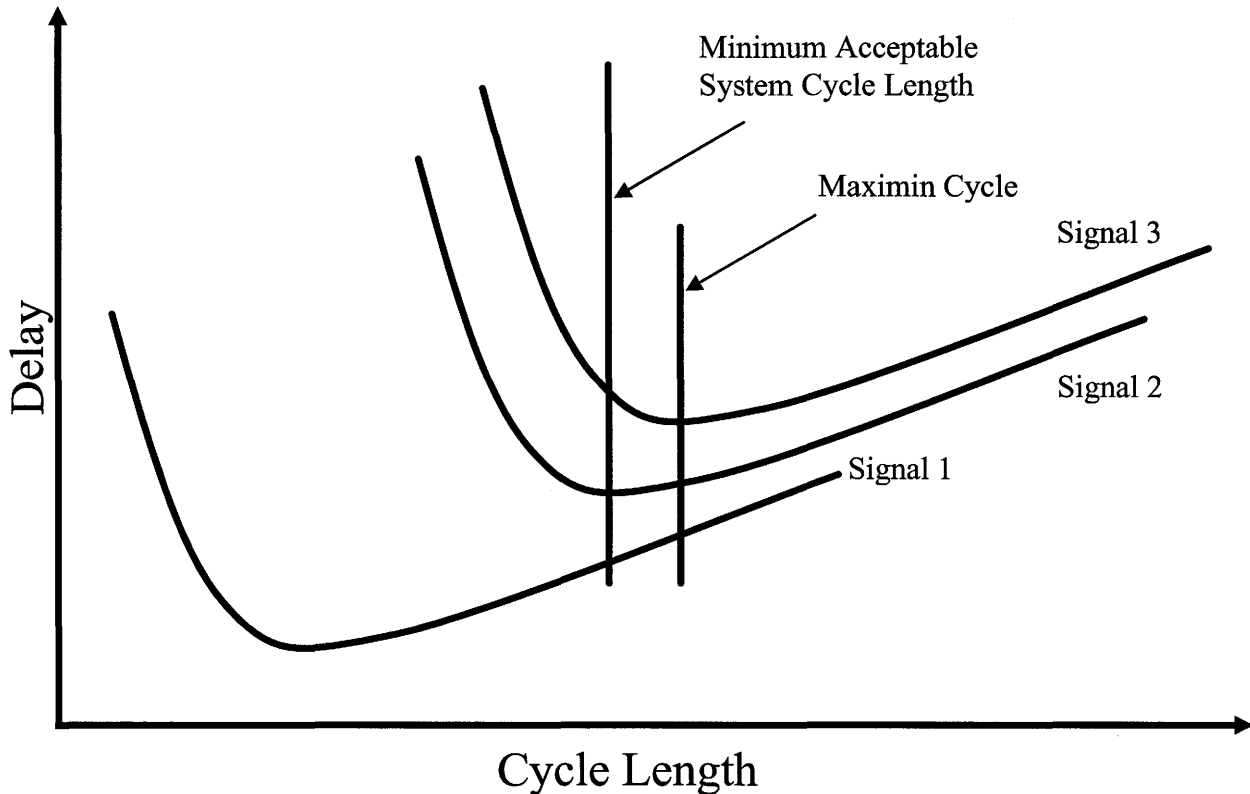
Figure 3. Half or Double Cycling Coordinating Traffic Signals.

For the case illustrated in Figure 3, assume that the *Maximin* cycle (corresponding to Signal 3) is 100 seconds, and the minimum-delay cycle length for Signal 1 is 55 seconds. Also assume that Signals 1, 2, and 3 are located on an arterial in that order. In this case, one has the following options:

- If traffic flow between Signal 1 and Signal 2 is insignificant or the distance between them is large, operate Signal 1 as isolated.
- Double-cycle Signal 1 and coordinate with the other signals.
- Operate all signals using a common cycle length constrained by the *Maximin* cycle length.

Once the analyst has selected a cycle length, green splits are generally determined by allocating the signal cycle according to volume-saturation flow ratios. However, splits must meet some minimum constraints based on pedestrian requirements and motorists' expectations. Furthermore, safety concerns for motorists may require precluding some phase patterns. An example of a safety concern is the use of split phasing at intersection approaches where the possibility of collision exists for opposing left-turn traffic.

Finding the best signal timing parameters for a signal system requires the evaluation of a selected objective function for all possible combinations of these parameters. The sheer number of these combinations makes it impractical to manually perform these calculations. Below, we provide some examples to illustrate the magnitude of the number of possibilities that need to be analyzed in order to select a coordination plan that best satisfies the selected objective.

Example 1: Assume a two-intersection arterial with two-phase signals and a fixed cycle length of 100 seconds. For this scenario, one needs to analyze the effects of 100 different offsets (0–99 seconds) on the selected objective.

Example 2: Adding another signal to the arterial in Example 1 will increase the possible combination of offsets to 200 if each link is assumed to be independent of the other, and 10,000 if one wishes to analyze the joint effects of changes in the offsets at both links.

Example 3: If all signals in the above example had multiple phases with protected left-turn phases, optimizing phase sequences for the two arterial directions will increase the number of possibilities by (4 × 4 × 4) 64 times (12,800 and 640,000 possibilities, respectively). In addition, if we desire to optimize phasing sequences on the cross-street approaches, the number of these possibilities will increase by another 64 fold.

Most real arterials have a significantly higher number of signals. In addition, most of the time, we also desire to optimize the cycle length. Thus, in many cases, there are millions of possible combinations of signal timing parameters based on the variables selected for optimization and the level of analysis detail. Fortunately, a number of computer programs are available to the traffic engineering community for use in such analyses. A majority of these programs use methods to minimize the number of combinations analyzed. Three of these computer programs will be described in this chapter.

Objectives of Signal Coordination

Earlier, we listed several objectives of signal coordination. Not all objectives apply to a given situation. Even when multiple objectives are applicable, selecting a timing plan to achieve one objective may not guarantee the achievement of other objectives. For instance, minimizing systemwide delay generally results in smaller queues, but not necessarily maximum progression bandwidths, and vice versa. Maximizing progression bandwidth, on the other hand, will generally result in the least number of stops, as well. However, it is possible to optimize signal timings to achieve one primary objective and then fine-tuning some signal timing parameters to improve a secondary objective without sacrificing the primary objective.

ANALYSIS AND OPTIMIZATION SOFTWARE

A number of computer programs are available to assist in the analysis and coordination of traffic signals on an arterial. All of these tools are based on the abstraction of reality and have their inherent weaknesses and strengths. In this section we provide a description of some commonly used programs for analyzing and optimizing arterial signal timings. We begin with the

description of a number of key concepts that will provide a better understanding of each program described in the following subsections.

Software for use in the analysis and/or optimization of isolated or coordinated signal timings may contain one or both of the following modules:

- a traffic simulation model, also called a traffic model; and
- an optimization model.

In the following subsections, we present a brief discussion of key concepts related to the above two types of models. This discussion is restricted to models for signalized arterials.

Traffic Signal Analysis Models

A traffic model takes traffic volumes, geometric information for the facility, and a complete description of a traffic control plan as input. Then, it simulates the described scenario and outputs measures-of-effectiveness (MOEs). Typical MOEs include: average or total delay, stops, fuel consumption, bandwidth efficiency, average or maximum queues, etc. Most models provide an estimate of several, if not all, MOEs. One method of model classification is the primary MOE used in the model. The two main types of models are: delay-based and bandwidth-based. Furthermore, the level of detail or abstraction used by a model is another classification method. There are three common types of traffic models based on the latter classification: microscopic, mesoscopic, and macroscopic.

Microscopic Traffic Models

Microscopic traffic models provide the most detailed analysis by simulating the behavior (acceleration, deceleration, car-following, etc.) of individual vehicles in the traffic stream. In general, these models are also stochastic in nature and rely on a random number generator that uses a seed value to generate values of various parameters during simulation. To obtain another sample, the user must change the seed value and re-run the simulation. Running the simulation with different random number seeds is equivalent to collecting a random sample of data, similar to collecting data for a peak period during many consecutive days. Due to the level of detail simulated, these models require the maximum amount of data and are the most computationally intense.

Mesoscopic Traffic Models

These models simulate traffic flow in specified time steps, and they are deterministic in nature. The time step can be 1 second, 2 seconds, or higher. For each time step, these models estimate the flow of traffic entering a link, traveling downstream, stopping due to a red light, and moving again when the light turns green. Some of these models also account for platoon dispersion as vehicles travel from one point to a downstream point in space. Mesoscopic models can be further classified as link-based or time-based. Link-based models simulate traffic flow one link at a time for all time steps in one signal cycle. These models treat a queue of vehicles at the signal approach as an upward stack. As a result, all vehicles arriving during red travel to the stopbar

and join a vertical (upward stack) queue. Link-based models cannot account for queue spillback because they do not keep track of the back of the queue. In addition, they may allow more vehicles to stack in a queue than a link's storage capacity. Thus, these models are not suitable for congested conditions or for short links where sub-optimal timing may cause queues to block flow from the upstream signal. Step-based models, on the other hand, simulate traffic flow on all links at each time step. These models can accurately account for the behavior of queued traffic and traffic flow interactions between adjacent links, and they are better suited for all types of traffic conditions in signal systems. The downside is that they are also more intense from a computational point of view. Also, the accuracy of these models may depend on the number of cycles simulated.

Macroscopic Traffic Models

Models in this category simulate the cycle-by-cycle behavior of platoons of traffic at each link in the system and are deterministic in nature. These models may or may not account for platoon dispersion. Macroscopic models treat a queue of vehicles at an approach as an upward stack. Thus, they are accurate only for undersaturated flow conditions. Because of their simplistic nature, macroscopic models are the most efficient from a computational point of view.

Optimization Models and Search Algorithms

As mentioned earlier, traffic models simulate a given set of traffic and control conditions. In other words, they are able to only tell how good or bad a given scenario is. Optimization and search algorithms are systematic techniques that systematically generate scenarios, compare their fitness or objective function value (i.e., delay, bandwidth efficiency, throughput, etc.) obtained by using a simulation model, and select the best scenario based on a predetermined criterion. For instance, if delay minimization is the desired objective, the primary fitness value will be the delay to motorists resulting from a specific scenario. Such an optimization model will evaluate the delay value for each alternative timing plan and select the timing plan that results in the least amount of delay. In other words, search algorithms are wrappers around traffic simulation models to provide the optimization function. Search algorithms can be simple or extremely sophisticated. Some of the common search algorithms are described below.

Exhaustive Search Algorithms

As the name implies, these algorithms calculate and compare the fitness values for all possible signal-timing scenarios. As illustrated previously (Examples 1–3), there can be millions of such combinations of signal timing parameters, depending on the size of the facility and how many variables are to be optimized simultaneously. Thus, an exhaustive search may require hours of computer time. Unless a model is designed for small facilities, the sheer number of possible scenarios usually requires the use of a divide-and-conquer strategy. For instance, computational time can be drastically reduced by stage-wise optimization of each variable instead of all variables simultaneously. Such a strategy increases computational efficiency by sacrificing accuracy. The positive feature of exhaustive algorithms is that full information is available for each scenario. Most optimization algorithms use some level of exhaustive search combined with other search algorithms.

Hill-Climbing Algorithm

A hill-climbing (or valley descent) algorithm starts with one (base) scenario, either specified by the user, selected by the program using a fixed criterion, or selected randomly. Then, it selects one or more variables (i.e., offset, cycle length, etc.). It creates two additional scenarios for each variable, one by increasing the values of that variable and the other by decreasing the value. The values of the selected variables are increased or decreased by specified amounts called step sizes. Following this, the algorithm uses a traffic simulator to calculate the fitness values for each of the two new scenarios and compares them with the base scenario. The scenario with the best fitness value identifies the two best scenarios and, consequently, a direction of further search. For instance, if increasing the value of the selected variable resulted in a better fitness value, the search algorithm will mark this new scenario as the current best and continue in the direction of increasing values for the variable. In the next iteration, the search algorithm generates a new scenario by increasing or decreasing the values of the selected variable in the selected search direction, calculating the new fitness value, and comparing it with the two current best values. The algorithm continues in this manner until the fitness value for the new scenario ceases to be better than the current best. Hill-climbing methods guarantee optimal solution only when the function to be optimized is unimodal (has one peak or valley). For multi-modal functions, the hill-climbing method may terminate with a suboptimal solution depending on how good the base scenario is. Most implementations of hill-climbing algorithms use sophisticated techniques, such as a variable step size, to speed up the search process.

Mathematical Programming Techniques

Mathematical programming techniques, such as linear- and integer-programming, require a complete specification of the objective (fitness) function along with all the applicable constraints of the traffic model in mathematical form (equations and/or inequalities) form. These techniques are based on systematic procedures (programs) that are designed to search a small subset of all possible scenarios in an intelligent manner. Mathematical programming techniques are applicable only when a closed-form mathematical model exists. When applicable, these techniques also guarantee the best solution. Further discussion of these techniques is beyond the scope of this report.

Genetic Algorithms

Genetic algorithms (GAs) belong to a class of algorithms known as evolutionary algorithms which have been developed fairly recently. A GA starts with a subset of scenarios (some members of a population) and applies principles of natural selection (mating, gene mutation, etc.) to generate a new or revised set of scenarios (called the next generation). A GA-based optimization model uses a specified traffic simulation model to evaluate the fitness of each member (i.e., a signal timing scenario) in the current population. Then, it generates a new population by combining the characteristics of (that is, by mating) selected pairs of scenarios (members). The principles of natural selection ensure that the characteristics of the fittest members (i.e., those with higher bandwidths or lowest delays, depending on the objective of optimization) have a high probability of transmission to the next generation. A GA terminates when either no more improvements occur, or a certain number of user-specified generations are

complete, whichever occurs first. GAs are different from all previously described search algorithms in that they utilize codings of variables rather than the values of variables. Given a large enough population and sufficient number of generations, a GA can provide the global optimum. This is because GAs perform simultaneous optimization of all variables. GAs can be applied to all types of optimization problems, even those that cannot be described in closed forms. Their effectiveness depends on the scheme used for coding the variables and the details of the natural selection process used.

Popular Signal Timing Optimization Programs for Arterials

Corridor Simulation Program

Corridor Simulation program (CORSIM) is a microscopic-stochastic simulation program (*3*). It has two modules: Freeway simulation (FRESIM) for evaluating freeway traffic conditions and network Simulation (NETSIM) for evaluating the quality of a selected signal timing plan. Traffic Viewer (TRAFVU) is an accompanying graphic animation program. NETSIM can be used to analyze the operation of pretimed and actuated signals. For a given scenario, CORSIM randomly generates traffic, keeps track of individual vehicles as long as they are in the system, and computes various measures of effectiveness (delay, stops, travel times, fuel consumption, etc.). Making a simulation-run using CORSIM is similar to one-time data collection in the field, for instance the duration of AM-peak period on Monday. Thus, it is necessary to make several runs using different random number seeds and averaging the results from those runs before drawing any conclusions. CORSIM was developed using Federal Highway Administration (FHWA) support over a period of several decades and is accepted by the transportation professionals as a valid analysis tool. CORSIM does not provide an optimization routine. Therefore, it is difficult, if not impossible, to use CORSIM for developing optimal signal timing plans.

Traffic Network Study Tool

Traffic Network Study Tool (TRANSYT) 7F is a mesoscopic-deterministic model for analyzing and optimizing signal timings on arterials and networks (*4*). Like CORSIM, TRANSYT 7F has been developed and tested over a period of several decades and has gained acceptance from the user community as a sound model. TRANSYT 7F uses a combination of exhaustive, hill-climbing, and GA-based optimization methods. TRANSYT 7F uses a delay-based traffic model. In other words, it is primarily designed to select signal timings that produce minimum system delay and stops. In addition, it provides a capability to select several secondary objectives, including minimization of stops and maximization of progression opportunities. During its optimization process, TRANSYT 7F generates second-by-second flow profiles of vehicles on all links in the network. Then, it analyzes these profiles to determine MOEs. TRANSYT 7F has two delay-based traffic models. The first model (original model) performs the optimization in a link-wise fashion by optimizing timings for one link at a time. This model does not accurately account for queue buildup because it treats a queue of vehicles as an upward stack at the stopbar. However, it works well for undersaturated traffic conditions. This model has been extensively validated by users all over the world. The second model was recently added to remove the limitations of the first model. This model takes into consideration the formation and dissipation

9

of queues in space. In addition, it accounts for flow interactions on adjacent links through a step-by-step analysis of all links in the system. Conceptually, this model is better suited for the analysis and optimization of congested (oversaturated) facilities. Consequently, it also requires more computation time. Until recently, the main deficiency of TRANSYT 7F has been its inability to optimize signal phase sequences. In the latest version of TRANSYT 7F (version 9) released earlier this year, this deficiency was removed through the addition of a GA-based optimization algorithm. TRANSYT 7F models actuated signals as equivalent pretimed signals, and it has the ability to half/double cycle traffic signals. Unfortunately, this version was not released in time for use in this project.

TRANSYT 7F performs exhaustive searches for cycle length. For each cycle, it starts by calculating equal saturation splits and applies a hill-climbing method to optimize signal offsets and splits. For this reason, its final results depend on the base timing plan supplied by the user. Although it contains a good delay-based traffic model, TRANSYT 7F's bandwidth analysis model is not very good. Last, but not least, learning to use TRANSYT 7F requires considerable effort. For these reasons, many practitioners dealing with signalized arterials, especially in Texas, prefer not to use it.

Synchro

Synchro (*5*) is a delay-based program for analyzing and optimizing timing plans for arterial and networks. Its objective function also minimizes stops and queue size by applying penalties for these measures. Synchro's traffic model is similar to the link-based model in TRANSYT 7F. Synchro uses an exhaustive search technique to optimize signal timings. To reduce the number of scenarios analyzed for a coordinated system, it relies on the divide-and-conquer principle. To optimize timings for an arterial, the program requires the user to apply several manual steps (cycle length optimization followed by offset and phase sequence optimization) in a specific order. It optimizes cycle length by analyzing all cycles in the defined range. Synchro optimizes offsets using a multi-stage process. At each stage, it uses a different step-size depending on the optimization level selected by the user. For instance, if the user requests extensive offset optimization, Synchro first simulates all offsets in 4-second increments, followed by a search using 2-second increments. Finally, it performs another search using 1-second increments in the vicinity of the best offset from the second stage. Unlike TRANSYT 7F, Synchro's traffic model does not consider platoon dispersion. As an alternate, it recommends when to coordinate two adjacent signals by calculating a *coordinatability factor* using link distance, travel time, and traffic volumes as input. Also, unlike other programs, Synchro generates optimal signal timings for each signal by averaging the analysis results of five volume scenarios for that signal. For this purpose, it assumes that a volume entered by the user is the mean and variance of the real traffic volume (Poisson distribution). Then, it applies factors from a normal distribution to generate four additional volume scenarios representing minus-2 (10th percentile), minus-1 (30th percentile), 1 (70th percentile), and 2 (90th percentile) standard deviations from the mean. In this scheme, user-supplied volumes are treated as 50th percentile volumes. In Synchro terminology, delay calculation based on this averaging method is referred to as the percentile delay method. Using this method, Synchro incorporates a method to model phase gapping and skipping behavior for actuated and actuated-coordinated signals. Synchro has, by far, the best user interface of all signal-timing tools currently available to the traffic-engineering professionals. It provides

features to easily fine-tune a timing plan. Furthermore, it provides for data conversion to other popular software. Due to this, Synchro popularity has grown at a phenomenal rate since its initial availability during the mid-1990s. However, since detailed analysis of the quality of its optimization and analysis methods has not been conducted by the traffic-engineering community, many practitioners are reluctant to use Synchro-generated timing plans in the field. Many engineers use it as an input processor for TRANSYT and CORSIM.

Progression Analysis and Signal System Evaluation Routine

Progression Analysis and Signal System Evaluation Routine (PASSER) II is a bandwidth-based program for optimizing signal timings for signalized arterials (*6*). Originally developed for TxDOT about 30 years ago, it has been one of the most popular programs in its class. The heuristic signal-timing optimization model of PASSER II is based on a graphical technique, and is simple, efficient, and powerful (*7*). PASSER II has passed the test of time and is known to produce good signal-timing plans. PASSER II can determine all four signal-timing variables described earlier. It selects the plan that maximizes progression efficiency, a unit-less quantity obtained by dividing the progression band by the cycle length. Because of its simplicity, it is also the most computationally efficient program in its class. PASSER II performs exhaustive searches over the range of cycle length provided by the user. It starts by calculating splits using Webster's method. Then, it applies a hill-climbing approach and adjusts splits to minimize delay. Finally, it applies its bandwidth optimization algorithm using the pre-calculated splits as input to that model. At the optimization stage, it can find the cycle length, offsets, and phase sequences that produce maximum two-way progression.

RESEARCH OBJECTIVES AND SCOPE

This two-year research project, funded by Texas Department of Transportation (TxDOT), had two primary tasks. The objective of Task 1 was to develop guidelines for using popular programs for timing signalized arterials. The bulk of this task was scheduled to be conducted during the first six months of the project. In this task, we used CORSIM 4.32 to compare the performances of Synchro 4, TRANSYT 7F (version 8.3), and PASSER II-90 for timing signalized arterial. These were the latest versions of these programs available at that time. The use of CORSIM provided an unbiased mechanism for comparison. Task 2 was to identify improvements needed in PASSER II and to upgrade its user interface, its optimization routine, and its traffic model.

ORGANIZATION OF THIS REPORT

The next chapter describes the software comparisons conducted in this project and provides research findings. In Chapter 3, we describe the enhanced PASSER V program developed in this project. This chapter also provides descriptions of new and enhanced models, followed by results of research conducted to validate the new models. In Chapter 4, we provide the results obtained by comparing PASSER V's new optimization models with the programs used in earlier software comparison studies. Chapter 5 provides conclusions and recommendations. Finally, the report includes several appendices to provide additional detail.

2. SOFTWARE COMPARISON STUDIES

INTRODUCTION

The first objective of this project was to compare the three most commonly used programs for timing signalized arterials and to develop guidelines for selecting one or more of these programs for arterial signal timing projects. These programs are: TRANSYT 7F, Synchro, and PASSER II. In this chapter, we describe the methodology used to conduct an unbiased analysis of these programs, data sets used for the analysis, and the three stages of analysis conducted in this research. Then, we present our findings.

RESEARCH METHODOLOGY

During the early stages of this project, researchers consulted the TxDOT advisory panel and made a decision to use real data for comparing Synchro, TRANSYT 7F, and PASSER II. Based on this decision, the researchers contacted several TxDOT districts, Texas cities, and consultants and requested data sets for use in this project. Many of these agencies favorably responded by supplying recent data they had used in signal-timing projects. From these data, the researchers selected two sets of arterials representing geometric and traffic characteristics found in most arteries in Texas. These characteristics include a range of:

- link lengths,
- number of lanes,
- lane assignments, and
- various cross street (one-way, two-way, and T) configurations.

The first data set consisted of five signalized arterials selected for use in the first two stages of analysis. One of these arteries has five intersections, and the others have six intersections each. We classified the original volume data for these arterials as the base volume scenario. Appendix A provides the sketches (Figures A1 through A5), base volumes and lane information (Tables A1 through A5), and link lengths (Table A6) for all of these arterials. In Tables A1-A5, the ">" and "<" symbols identify the cases where a left- or right-turn movement shared the lane with through traffic. For each of these arterials, we derived two additional volume scenarios — 70 and 120 percent — from the base volume data. This resulted in a total of 15 scenarios.

The second data set, selected for use in the third stage of analysis, consisted of three larger arterials with 10, 12, and 14 intersections, respectively. Figures A6 through A8 in Appendix A provide sketches of these arteries. Furthermore, two of these arterials had data available for both AM- and PM-peak periods, whereas one arterial had data available for only AM-peak. This resulted in a total of five volume-geometry scenarios. For brevity, we have omitted details of these data. Readers interested in those details are welcome to contact the authors.

For each stage of analysis, we used up-to-date versions of each program. The following subsections describe the three stages of analysis. We conducted the first two stages of analysis between September 2000 and March 2001. We conducted the third stage in 2002. The purpose of the third stage was to answer questions arising as a result of the previous analyses and to update

13

the findings by including newer versions of the two best programs. Furthermore, for all analyses, we selected the default optimization criteria of each program.

The First Two Stages of Analysis

The following list identifies the version of each program used and its features selected for the initial analysis:

- TRANSYT 7F, version 8.2. The researchers requested the program to use the step-wise model. All of these optimization runs consisted of a 5-minute initialization period and a 15-minute analysis period after that. Since TRANSYT 7F, version 8.2, cannot optimize phase sequences, we selected leading left-turn phases for all TRANSYT 7F runs.
- Synchro, version 4.0, Build 223. In Synchro, the researchers selected the percentile delay method and extensive offset optimization. In addition, we requested Synchro to not allow uncoordinated signals and double-cycled signals in the system.
- PASSER II-90, version 2.0. The researchers requested PASSER II to optimize all signal-timing parameters.

In order to ensure unbiased comparison of the three programs, we used the NETSIM component of CORSIM to simulate the timings generated by each program for all scenarios analyzed. Also, for each scenario, we conducted 20 replications of 15-minute simulations. Then, we averaged CORSIM replication results for each scenario and compared the following measures-of-effectiveness obtained from CORSIM:

- average seconds per vehicle delay for all vehicles in the system,
- average seconds per vehicle delay for arterial traffic only,
- system throughput in vehicles serviced per hour, and
- arterial throughput in vehicles serviced per hour.

In addition, we compared optimal cycle lengths and progression bandwidth efficiencies produced by the three programs for each scenario. Appendix B provides plots of these results. A summary of findings will be presented later in this chapter.

Scenarios Analyzed

As mentioned above, the researchers analyzed a total of 15 artery-volume scenarios. Due to time constraints the researchers restricted the analysis to pretimed signals and protected phases only. This analysis was divided into the following two stages:

1. During the first stage, the researchers requested each model to optimize the cycle length as well as all other parameters it is capable of optimizing. For this analysis, researchers selected a cycle length range of 40 to 150 seconds in 5-second increments. In other words, we requested each program to consider 13 different cycle lengths and to automatically select the best cycle length and timing parameters based on its default optimization criteria.

2. Stage 1 analysis revealed that the three programs selected significantly different optimal cycle lengths. Since a complete comparison of the three programs could not be conducted using these results, the researchers decided to conduct a more in-depth analysis using controlled cycle lengths. For this additional analysis, the researchers used the base volume scenario only. In this analysis, the researchers analyzed the optimization capabilities from each model for fixed cycle lengths within a carefully selected cycle length range. In other words, we made optimization runs by using fixed cycle lengths. For each arterial, we selected a cycle length range based on the *Maximin* criterion described in Chapter 1. Table 1 provides information about the cycle lengths considered in this analysis.

Table 1. Controlled Cycle Length Data for Stage 2 Analysis.

Artery Number	Artery Name	Cycle Lengths Analyzed
1	Arapaho	90, 100, 110, 120
2	FM 1709	90, 100, 110, 120
3	Belt Line	60, 70, 80, 90
4	FM 1179	90, 100, 110, 120
5	Frankford	90, 100, 100, 120

Third Stage of Analysis

In this stage of analysis, we compared Synchro 5 (Build 321) and PASSER II-02. These versions of the two programs became available after completion of the previous analysis, and they contain minor updates to the optimization/analysis technology in the previous versions. The analysis was similar to the second analysis stage, but for arterials much larger than those used previously. Table 2 provides information about the cycle lengths used for each artery. We selected the lower bounds of these ranges using Synchro's recommendation.

Table 2. Sixty-one Controlled Cycle Length Scenarios for Stage 3 Analysis.

Artery Number	Artery Name	Case	Cycle Lengths Analyzed
6	Stevens Creek	AM-Peak	95 to 150 with 5-second increments
		PM-Peak	95 to 150 with 5-second increments
7	De Anza Boulevard	AM-Peak	100 to 150 with 5-second increments
		PM-Peak	100 to150 with 5-second increments
8	US-90A	AM-Peak	80 to 150 with 5-second increments

As opposed to previous studies in which we used CORSIM, here we used each program to simulate the timings produced by the other program. Our objective was two-fold. First, we wanted to understand the performance of these two programs for larger arterials. Second, we wanted to find the differences in the MOEs produced by the two programs. In order to accomplish these objectives, we made the following sets of runs for the five cases and all selected cycle lengths for each case:

- optimized signal timings using Synchro,
- optimized signal timing using PASSSER II,

15

- simulated PASSER results using Synchro, and
- simulated Synchro results using PASSER II.

Then, we performed the following analysis for each of the 61 scenarios:

- compared MOEs for optimization solutions generated by each program;
- for each optimal solution from PASSER II, compared PASSER II MOEs against those produced by Synchro simulation; and
- for each optimal solution from Synchro, compared Synchro MOEs against those produced by PASSER II simulation.

Furthermore, we conducted additional analyses to answer additional concerns. These additional analyses were as follows:

1. We optimized a specific Synchro solution using PASSER II. In this case, we simulated Synchro splits, but allowed PASSER II to optimize offsets and phasing sequences.

2. We selected one intersection and applied five growth factors to obtain a range of very low- to very high-volume conditions. Then, we compared delay estimates from PASSER II and Synchro for a range of cycle lengths for each of the five volume scenarios. We studied thirteen cycle lengths between 40 and 150 seconds, inclusive.

SUMMARY OF FINDINGS

Appendix B contains graphs and tables providing detailed results from the first two stages of analysis. Appendix C contains the results of analyses from Stage 3.

Stage 1 Results

The objective of the first stage of analysis was to learn about the performance and behavior of each program in the absence of any user intervention.

Cycle Length

Table 3 provides information about the best cycle lengths selected by each program for the 15 scenarios.

Table 3. Optimal Cycle Lengths Selected by Synchro, PASSER II, and TRANSYT 7F.

Growth Factor	0.7 (70% of Base Volumes)			1.0 (Base Volumes)			1.2 (120% of Base Volumes)		
Artery	Synchro	PASSER	TRANSYT	Synchro	PASSER	TRANSYT	Synchro	PASSER	TRANSYT
1	50	135	75	70	140	120	75	150	130
2	65	106	105	80	106	95	95	106	95
3	50	145	40	75	145	45	70	145	65
4	65	110	75	85	110	60	90	110	60
5	65	120	110	100	120	100	100	120	100

16

From this table the reader can observe that the three programs selected different cycle lengths and, as a result, significantly different timing plans. To a certain extent, such differences are expected between delay-based and bandwidth-based optimization models. However, large differences between Synchro and TRANSYT 7F (both delay-based models) were unexpected. The following is a summary of general observations:

- In general, Synchro was consistent in its cycle length selection for volume growth factors of 0.7 (low), 1.0 (medium), and 1.2 (high). As the reader can see from Table 3, Synchro generally selected lower cycle lengths for low volumes and higher cycle lengths as the volumes increased.
- PASSER II generally selected higher cycle lengths. In addition, it had less or no variability for different volume scenarios for the same artery. This is not unexpected because bandwidth optimization depends more heavily on travel times. Travel time on a link is a function of link distance and speed, and both remained constant in our studies.
- TRANSYT 7F had the least consistency in its selection of cycle length. Sometimes, cycle length increased with an increase in volumes on an artery and at other times it decreased. The latter situation may be the result of TRANSYT 7F's action to minimize queue spillback.

Progression Bandwidth

Optimal results from the three programs showed that the timing plans generated by TRANSYT 7F had no progression bands. On the other hand, both PASSER II and Synchro produced timing plans that had through progression bands. Figures B1 through B3 in Appendix B provide comparisons of progression bandwidth efficiencies produced by these two programs. The following is a summary of the findings:

- PASSER II consistently produced solutions with better bandwidth efficiency than Synchro. In many cases, this difference was significant.
- For one case (Artery 4), Synchro produced bands almost the same as PASSER II.

Average Systemwide Delay per Vehicle

Figures B4 through B6 in Appendix B provide the comparison of average delay, in seconds per vehicle, for the three volume scenarios. The results can be summarized as follows:

- For all scenarios, Synchro produced the same or lower delays than TRANSYT 7F.
- For most (87 percent) cases, Synchro produced lower delays than PASSER II. For the remainder (13 percent) PASSER II produced lower delays.
- In 87 percent of the cases, PASSER II produced lower delays than TRANSYT 7F.

System Throughput

Throughput is the number of vehicles, in vehicles per hour (vph), serviced by the system. Thus, the higher the throughput, the better the timing plan. Figures B7 through B9 in Appendix B show the results of this analysis. The results are summarized below:

- The throughput for all timing plans from all three programs was similar.
- In 73 percent of the cases, throughput of PASSER II timing plans was the same or slightly higher than that for Synchro or TRANSYT 7F. In only one case, PASSER II resulted in lower throughput than both Synchro and TRANSYT 7F.
- In 87 percent of the cases, Synchro resulted in higher throughput than TRANSYT 7F.

Throughput for Arterial Traffic Only

Figures B10 through B12 in Appendix B present the results of this analysis. These results can be summarized as follows:

- For 80 percent of the cases, PASSER II solutions produce the same or better arterial throughput than Synchro and significantly more throughput than TRANSYT 7F.
- For only one of the cases, TRANSYT 7F throughput was significantly higher than PASSER II and Synchro.

Stage 2 Analysis

Figures B13 through B32 in Appendix B contain the results of this analysis. The following subsections provide a summary of each measure of effectiveness studied.

Progression Bandwidth Efficiency

As in the previous case, TRANSYT 7F did not produce any progression bands. Figures B13 through B17 in Appendix B provide results of this analysis. The following is a summary of the Synchro and PASSER II comparison:

- With one exception, PASSER II produced the same or larger bandwidth efficiency than Synchro.
- For 80 percent of the cases, PASSER II band efficiency was significantly higher than Synchro.
- For one case (70-second cycle length for Artery 3), PASSER II bandwidth efficiency was lower than Synchro.

Average Delay per Vehicle

Figures B18 through B22 in Appendix B provide the results of this analysis. The following is a summary of findings:

- Except one case, PASSER II delay was less than TRANSYT 7F.
- Except one case, Synchro produced lower delays than TRANSYT 7F.
- In 75 percent of the cases, PASSER II delay was less than Synchro.

System Throughput

Figures B23 through B27 in Appendix B provide the results of this analysis. The following is a summary of the findings:

- In general (90 percent of the cases), PASSER II produced higher system throughput than Synchro and TRANSYT 7F.
- In several cases, PASSER II throughput was significantly higher than the other two programs.
- In four cases, PASSER II throughput was lower than that of TRANSYT 7F. In one of these cases, PASSER II throughput was lower than Synchro.

Throughput for Arterial Traffic Only

Figures B28 through B32 in Appendix B present the results of this analysis. The following is a summary of findings:

- In most cases, PASSER II arterial throughput was higher than the other programs.
- In most cases, Synchro outperformed TRANSYT 7F.

Stage 3 Analysis

Tables 4 through 8 provide summaries of optimization runs from Synchro 5 and PASSER II-02. We have omitted summary tables for simulation runs from the two programs. However, the figures provided in Appendix C contain comparisons of all results. In the following subsections, we provide a summary of results.

Comparison of Bands for Synchro and PASSER II for Large Arterials

Figures C1 through C5 and Figures C6 through C10 in Appendix C provide comparisons of bandwidths and band efficiencies produced by Synchro and PASSER II for large arterials. The analysis shows that PASSER II provides the best two-way progression for large arterials, as well. Simulations of PASSER II timings using Synchro showed that bands were, for most cases, as good as those estimated by PASSER II. In a few cases, Synchro estimated slightly lower bands for PASSER timings. However, Synchro's ability to generate timings with good arterial progression significantly degraded for these large arterials. For these cases, we found that Synchro produced bands comparable to PASSER II in a few cases, only one-way progression in many cases, and no progression in some cases. In one case, (Stevens Creek PM-peak volumes at 140-second cycle length), Synchro found better progression than PASSER II. Total bands from these programs for this scenario were 52 and 44 seconds, respectively. This can happen because the two programs calculate different splits. To verify this hypothesis, we conducted the following two additional runs by requesting PASSER II to optimize:

1. offsets using Synchro splits and phase sequences, and
2. offsets and phase sequences using Synchro splits.

Table 4. Synchro 5.0 Optimization versus PASSER II Optimization for Stevens Creek AM-Peak Case.

Cycle Length (sec)		95	100	105	110	115	120	125	130	135	140	145	150
Synchro Optimization	delay (sec/veh)	20	18	19	18	20	20	20	21	22	22	23	24
	fuel consumption (g/h)	587	562	608	563	576	588	575	582	587	596	599	618
	EB band (sec)	0	0	0	0	0	0	7	9	0	3	13	0
	WB band (sec)	12	17	0	11	7	0	0	16	18	19	19	17
	sum of band (sec)	12	17	0	11	7	0	7	25	18	22	32	17
	efficiency (%)	6.3%	8.5%	0.0%	5.0%	3.0%	0.0%	2.8%	9.6%	6.7%	7.9%	11.0%	5.7%
	stops/hr	22749	21274	24844	21188	21570	22238	20941	21291	21349	21853	21579	22635
PASSER II Optimization	delay (sec/veh)	41	35	34	34	34	31	31	32	33	34	35	35
	fuel consumption (g/h)	727	665	630	625	621	603	604	598	607	609	614	619
	EB band (sec)	8	7	7	8	8	10	12	14	15	16	16	15
	WB band (sec)	11	9	10	11	11	14	16	18	20	21	21	20
	sum of band (sec)	19	16	17	19	19	24	28	32	35	37	37	35
	efficiency (%)	10.0%	8.0%	8.1%	8.6%	8.3%	10.0%	11.2%	12.3%	13.0%	13.2%	12.8%	11.7%
	stops/hr	32094	29924	27378	26713	26143	26240	26080	25260	25088	24858	24687	24661
PASSER II to Synchro Delay Ratio		2.05	1.94	1.79	1.89	1.70	1.55	1.48	1.45	1.50	1.55	1.52	1.46

Table 5. Synchro 5.0 Optimization versus PASSER II Optimization for Stevens Creek PM-Peak Case.

Cycle Length (sec)		95	100	105	110	115	120	125	130	135	140	145	150
Synchro Optimization	delay (sec/veh)	24	23	22	22	22	23	24	24	24	24	25	26
	fuel consumption (g/h)	856	839	826	818	809	837	865	844	860	843	846	866
	EB band (sec)	1	15	13	0	0	0	0	0	19	24	1	27
	WB band (sec)	4	12	0	8	0	17	0	9	0	28	22	0
	sum of band (sec)	5	27	13	8	0	17	0	9	19	52	23	27
	efficiency (%)	2.6%	13.5%	6.2%	3.6%	0.0%	7.1%	0.0%	3.5%	7.0%	18.6%	7.9%	9.0%
	stops/hr	31203	31164	30653	29950	29080	31200	32790	30855	32020	30877	30138	31407
PASSER II Optimization	delay (sec/veh)	54	48	45	41	38	38	40	40	39	40	40	41
	fuel consumption (g/h)	1113	1041	998	938	904	904	911	910	887	896	899	904
	EB band (sec)	18	18	17	17	17	18	19	22	24	23	24	26
	WB band (sec)	17	16	15	15	16	17	17	20	22	21	22	24
	sum of band (sec)	35	34	32	32	33	35	36	42	46	44	46	50
	efficiency (%)	18.4%	17.0%	15.2%	14.5%	14.3%	14.6%	14.4%	16.2%	17.0%	15.7%	15.9%	16.7%
	stops/hr	44360	42059	40651	38107	37094	37387	36206	36280	35284	35153	34624	34361
PASSER II to Synchro Delay Ratio		2.25	2.09	2.05	1.86	1.73	1.65	1.67	1.67	1.63	1.67	1.60	1.58

Table 6. Synchro 5.0 Optimization versus PASSER II Optimization for De Anza Blvd. AM-Peak Case.

Cycle Length (sec)		100	105	110	115	120	125	130	135	140	145	150
Synchro Optimization	delay (sec/veh)	21	20	21	20	21	21	22	22	23	24	24
	fuel consumption (g/h)	1151	1152	1113	1111	1106	1106	1137	1126	1146	1151	1159
	EB band (sec)	0	0	0	12	0	0	0	0	0	0	0
	WB band (sec)	17	20	30	26	29	24	21	19	18	11	29
	sum of band (sec)	17	20	30	38	29	24	21	19	18	11	29
	efficiency (%)	8.5%	9.5%	13.6%	16.5%	12.1%	9.6%	8.1%	7.0%	6.4%	3.8%	9.7%
	stops/hr	43775	44163	39964	41031	40018	39408	41243	40326	40971	40571	40820
PASSER II Optimization	delay (sec/veh)	84	70	62	56	53	48	45	42	41	43	42
	fuel consumption (g/h)	2102	1901	1783	1674	1622	1523	1470	1425	1416	1410	1394
	EB band (sec)	10	12	11	11	12	12	12	14	14	15	16
	WB band (sec)	23	26	24	25	27	27	27	31	32	34	37
	sum of band (sec)	33	38	35	36	39	39	39	45	46	49	53
	efficiency (%)	16.5%	18.1%	15.9%	15.7%	16.3%	15.6%	15.0%	16.7%	16.4%	16.9%	17.7%
	stops/hr	63686	61436	59333	56657	55280	51760	50297	49757	50148	47571	47564
PASSER II to Synchro Delay Ratio		4.00	3.50	2.95	2.80	2.52	2.29	2.05	1.91	1.78	1.79	1.75

Table 7. Synchro 5.0 Optimization versus P2 Optimization for De Anza Blvd. PM-Peak Case.

Cycle Length (sec)		100	105	110	115	120	125	130	135	140	145	150
Synchro Optimization	delay (sec/veh)	25	25	24	23	23	23	22	23	23	23	23
	fuel consumption (g/h)	1299	1308	1257	1248	1251	1263	1229	1226	1216	1230	1228
	EB band (sec)	35	36	27	32	29	14	27	29	0	20	4
	WB band (sec)	0	0	0	0	0	0	0	0	12	0	15
	sum of band (sec)	35	36	27	32	29	14	27	29	12	20	19
	efficiency (%)	17.5%	17.1%	12.3%	13.9%	12.1%	5.6%	10.4%	10.7%	4.3%	6.9%	6.3%
	stops/hr	45150	46011	42778	43185	43541	44496	42265	41690	40499	41476	40988
PASSER II Optimization	delay (sec/veh)	78	68	56	48	43	42	40	39	38	38	36
	fuel consumption (g/h)	2299	2093	1910	1759	1662	1612	1566	1539	1531	1511	1460
	EB band (sec)	21	25	27	27	26	30	32	31	41	40	41
	WB band (sec)	11	13	14	14	14	16	17	16	22	21	22
	sum of band (sec)	32	38	41	41	40	46	49	47	63	61	63
	efficiency (%)	16.0%	18.1%	18.6%	17.8%	16.7%	18.4%	18.8%	17.4%	22.5%	21.0%	21.0%
	stops/hr	74591	68868	66443	62845	60286	57757	55637	54498	55216	53104	50992
PASSER II to Synchro Delay Ratio		3.12	2.72	2.33	2.09	1.87	1.83	1.82	1.70	1.65	1.65	1.57

Table 8. Synchro 5.0 Optimization versus P2 Optimization for US-90A.

Cycle Length (sec)		80	85	90	95	100	105	110	115	120	125	130	135	140	145	150
Synchro Optimization	delay (sec/veh)	93	88	84	86	79	81	82	79	82	85	77	80	80	83	79
	fuel consumption (g/h)	2129	2005	1939	1939	1831	1831	1856	1817	1815	1865	1738	1775	1748	1770	1723
	EB band (sec)	0	0	0	16	8	0	1	16	0	0	0	0	0	0	0
	WB band (sec)	0	0	10	15	15	0	0	0	0	0	20	6	26	25	0
	sum of band (sec)	0	0	10	31	23	0	1	16	0	0	20	6	26	25	0
	efficiency (%)	0.0%	0.0%	5.6%	16.3%	11.5%	0.0%	0.5%	7.0%	0.0%	0.0%	7.7%	2.2%	9.3%	8.6%	0.0%
	stops/hr	69792	63120	59602	58771	53913	52892	53423	51990	50197	51499	46329	47254	45159	44647	43510
PASSER II Optimization	delay (sec/veh)	146	132	124	117	114	110	106	104	103	101	100	98	97	95	95
	fuel consumption (g/h)	2467	2281	2171	2064	2009	1960	1894	1867	1823	1805	1781	1746	1735	1711	1705
	EB band (sec)	10	14	17	18	16	23	27	28	30	30	29	32	34	35	36
	WB band (sec)	14	19	24	25	22	31	34	36	38	38	40	42	44	45	46
	sum of band (sec)	24	33	41	43	38	54	61	64	68	68	69	74	78	80	82
	efficiency (%)	15.0%	19.4%	22.8%	22.6%	19.0%	25.7%	27.7%	27.8%	28.3%	27.2%	26.5%	27.4%	27.9%	27.6%	27.3%
	stops/hr	75947	69834	65802	62019	59430	57592	55401	53958	51340	50491	49384	47731	47516	46544	45984
PASSER II to Synchro Delay Ratio		1.57	1.50	1.48	1.36	1.44	1.36	1.29	1.32	1.26	1.19	1.30	1.23	1.21	1.14	1.20

From the results, we found that PASSER II provided successively better progression band than Synchro for the two scenarios. Total PASSER II band from the second scenario was 62 seconds, a significant improvement over Synchro. This reinforces the fact that PASSER II does provide the best bandwidth optimization model, whereas, Synchro fails to consistently produce good bands for large arterials. This result is expected because Synchro is not designed to explicitly search for good bands.

Delay Comparison of Synchro and PASSER II

Figures C11 through C30 in Appendix C compare delays produced by each program for the five scenarios. Figures C11 through C15 in Appendix C compare delays produced by each program for the timing plans it generated. Numbers in the plots provide the ratios of PASSER II to Synchro delay for each plan. As the figures show, PASSER II produced significantly higher delays than Synchro for all scenarios. However, a direct comparison of these results is not appropriate because there is no common basis for comparison. Recall that previous comparisons of timings using CORSIM as a common basis did not show this magnitude of difference. A more suitable comparison of these graphs would be to study the trends in delay estimates. From these plots, we observed that PASSER II's delay estimates have a more pronounced similarity to Webster's delay-versus-cycle length curve, while Synchro's delays were almost horizontal. This may be because we used Synchro to determine the lower bound of the cycle length range, which is Synchro's estimated minimum-delay cycle. From these plots, we also observed that PASSER II minimum delay occurs at a higher cycle length than Synchro. Lastly, the differences between the programs seem to reduce for larger cycle lengths. Figures C16 through C20 in Appendix C show assessments of PASSER II timings by PASSER II and Synchro. Figures C21 through C25 in Appendix C show assessments of Synchro timing plans by both programs. From these, we conclude that the Synchro model estimates significantly lower delays than PASSER II for the same timing plan and provides solutions with lower delay than PASSER II. These comparisons cannot reveal which model is better and closer to reality. Finally, Figures C26 through C30 in Appendix C show a comparison of Synchro timing plans with PASSER II timing plans, both assessed by Synchro. These comparisons show that Synchro produces timings with lower delay than PASSER II.

In order to learn more about the delay models in the two programs, we needed additional studies. We took a single intersection (Intersection Stevens Creek and De Anza Blvd.), created several low (GF 0.5) to high (GF 1.5) volume scenarios by applying growth factors, and optimized these scenarios over a range of cycle lengths using each program. The results are provided in Figures C31 through C37 in Appendix C. From these runs, we learned the following for isolated signals:

1. The Synchro model estimates lower delays than the PASSER II model.
2. For low-volume conditions, the results are close, but the gap starts increasing as volumes increase.
3. Synchro's estimate of minimum delay cycle length is lower than PASSER II's estimate.

SUMMARY AND CONCLUSIONS

From extensive comparisons of Synchro, TRANSYT 7F, and PASSER II, we found the following:

- PASSER II and Synchro outperform TRANSYT 7F in all aspects.
- PASSER II consistently produced solutions with better progression bandwidth efficiency and throughput than Synchro. Synchro produces good progression for small arterials, but its ability to produce two-way progression degrades for larger arterials.
- Synchro produced timings with the lowest delays, but PASSER II can produce comparable delays if its search is guided using a good cycle length, especially for small arterials.

Based on these findings, we offer the following guidelines and suggestions for selecting software in arterial signal-timing projects:

1. Do not use TRANSYT 7F.
2. Use PASSER II when it is important to provide maximum arterial progression and throughput.
3. Use Synchro when minimizing delay has higher priority than optimizing arterial progression.

PASSER II-02 saves timing plans for all cycle lengths analyzed and provides a summary table of MOEs corresponding to the best plan for each cycle length. The user should select a timing plan for the cycle length that provides maximum progression efficiency and lower delay. Also, when a user desires to obtain good progression bands from Synchro, he or she should run the program for a range of cycle lengths, review the time-space diagrams for each solution, and select the cycle length that results in the best progression. This is because the best timing plan selected by Synchro may not provide any progression.

3. PASSER ENHANCEMENTS

Comparison of three popular programs for timing arterials revealed that only PASSER II among these programs guarantees two-way arterial progression on signalized arterials. Our studies showed that bandwidth-based timings produce fewer stops and higher productivity. In addition, the studies showed that engineering judgment can be used to select bandwidth-based timing plans that also produce lower, if not always lowest, vehicular delays. In addition, bandwidth-based optimal timing plans for arterials are also easily recognized and appreciated by drivers. Therefore, we concluded that we should retain PASSER II's bandwidth optimization model in the new software. In this chapter, we provide detailed information about the version of PASSER developed in this project. We developed the new program using Borland C++ Builder 5.0 (*8*) and named it PASSER V. This change in name reflects the fact that the new program includes features above and beyond those required by the TxDOT contract.

OPTIMIZATION ALGORITHMS IN PASSER V

Interference Minimization Algorithm

This algorithm is a revised version of the optimization algorithm used by PASSER II. Here, we describe the PASSER II implementation of the interference minimization algorithm and then describe modifications for implementation in PASSER V.

Like most programs, PASSER II calculates preliminary splits for each signal based on Webster's method. Then, PASSER II applies an optimization method to adjust these splits to minimize intersection delay. These pre-calculated splits are then input to the bandwidth optimization algorithm. For bandwidth optimization, PASSER starts by selecting a cycle and calculates perfect one-way progression in the A (arbitrarily selected) direction. Then, it minimizes band interference in the B (opposite) direction by adjusting phasing sequences and offsets. The maximum total band calculated by the program is as follows:

$$Total\ Band = G_A + G_B - I \qquad\qquad \text{Equation 4}$$

where:

G_A = Least green in A-direction;
G_B = Least green in B-direction; and
I = Minimum possible band interference.

After achieving the best band (minimum interference) in the B direction, the program adjusts the two bands according to user-desired options for directional priority. The reader should note here that the interference minimization algorithm intelligently searches a very small subset of all possible combinations of signal timings. Finally, the program calculates delays, bandwidth efficiency, and attainability. Delay calculation is based on a macroscopic traffic model. Efficiency and attainability measure how good a bandwidth solution is. Efficiency for a direction is the percent of cycle used for progression. Attainability is the percent of bandwidth in a direction in relation to the minimum green split in that direction. Theoretically, the maximum bandwidth in a direction can be no more than the smallest through green split in that direction.

The following formulas are used to calculate combined efficiency and attainability for the two arterial directions:

$$Progression\ Efficiency\ (\%) = \frac{(Arterial\ Band_A + Arterial\ Band_B)}{2*Cycle\ Length} * 100 \qquad \text{Equation 5}$$

$$Progression\ Attainability\ (\%) = \frac{(Arterial\ Band_A + Arterial\ Band_B)}{(Min.Green_A + Min.Green_B)} * 100 \qquad \text{Equation 6}$$

The reader should note that while bandwidth generally increases with an increase in cycle length, efficiency may increase, decrease, or remain constant. Thus, it is desirable to select a solution that provides the best efficiency and an attainability of 100 percent. In addition, the timing plan should not use cycle lengths larger than that necessary to move traffic through all approaches on the arterial.

Research in this project revealed that PASSER II has a tendency to select larger cycle lengths. Upon conducting further research, we found that this tendency was because of the split optimization feature implemented in PASSER II. What happens in PASSER II is that the split optimization gives more and more green time to the through traffic as cycle lengths increase. Since PASSER II optimized bandwidth, it tends to select larger cycle lengths because of larger bands. Thus, we decided to not implement this feature in the PASSER V implementation of the algorithm. Thus, the splits calculated in PASSER V are slightly different from those in PASSER II. In addition, we found that in some cases, the interference minimization algorithm of PASSER II ends before finding the best solution. The reasons for this result are the heuristic nature of the algorithm and the fact that the algorithm only considers a subset of all possible solutions. In the PASSER V implementation, we apply the algorithm for both directions, thereby increasing the ability of the algorithm to find better solutions.

Exhaustive Search

As explained earlier, the exhaustive search methods evaluate all possible combinations of selected signal timing variables. This method of optimization works well for small problems. In PASSER V, the *PASSER II* tool uses exhaustive search method for cycle length search, and the *PASSER III* tool uses this method for all variables in isolated diamond interchanges. The *PASSER III* tool provides features similar in function to the PASSER III program (*9*).

Genetic Algorithm

PASSER V uses a genetic algorithm to provide new features to develop signal timings for minimizing delay or for maximizing arterial progression. Here, we provide a general description of GAs. As stated in the first chapter, GAs are optimization techniques based on the concepts of natural selection and genetics. Genetic algorithms differ from traditional algorithms in that they work with a coding of the parameter set, not the parameters themselves; search from a population

of points, not a single point; and use probabilistic rules, not deterministic rules. In the genetic algorithm approach, the variables are represented as genes of a chromosome. The standard genetic algorithm proceeds as follows (*10, 11, 12*):

1. It randomly or heuristically generates an initial population (generation 0) of candidate solutions for a given problem.
2. For every evolutionary step known as a generation, it evaluates the fitness (bandwidth, delay, etc.) of each solution.
3. It forms a new population (the next generation) by selecting the individuals with best fitness and applying natural selection schemes (genetic operation, mutation, and recombination) to pairs of individuals.
4. It deletes unwanted members of the population to make room for new members.
5. It evaluates new individuals and inserts them into the population pool.
6. If termination criterion is met, it stops, otherwise it goes back to step 3.

One iteration of this loop is referred to as a generation. Natural selection guarantees that individuals with the best fitness will propagate into future populations. Using the recombination operator, the GA combines genes from two parents to form two new offspring that have a high probability of having better fitness than their parents. Mutation allows infusion of features not present in parents. Over several generations, the best individuals survive and the worst are eradicated. Figure 4 shows a flow chart of this methodology.
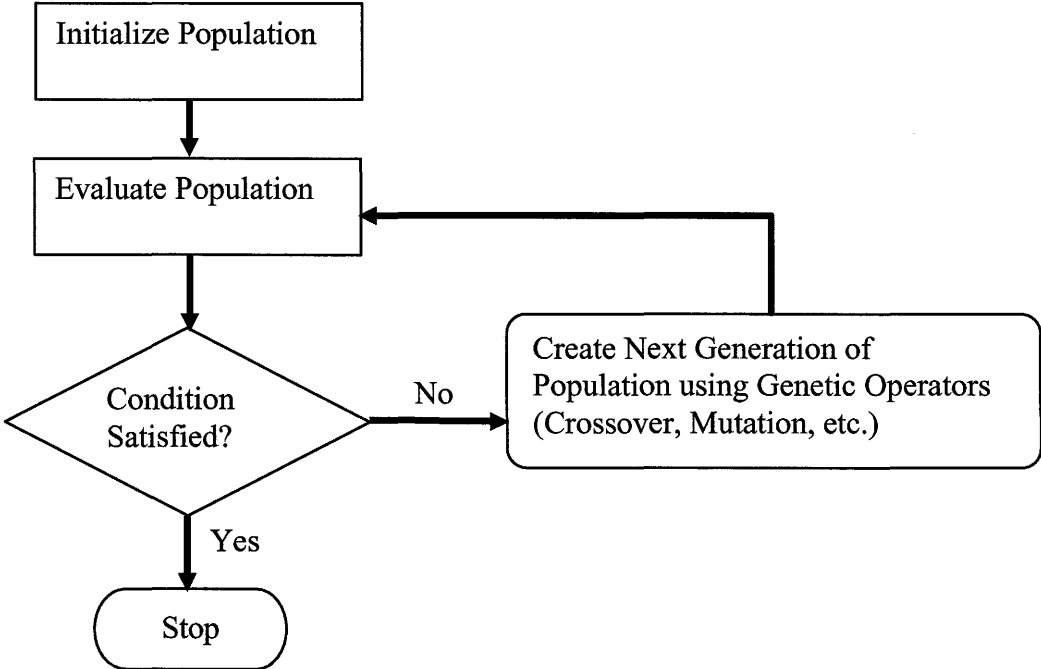
Figure 4. The Structure of a Genetic Algorithm.

Genetic algorithms provide the capability of optimizing all desired signal timing parameters in parallel, unlike the hill-climbing method, which optimizes one timing parameter at a time. Consequently, GAs may also require more time. Many studies conducted to date have shown that GA-based optimization performs better than the hill-climbing method.

A GA software, or driver, must be employed for applying this optimization technique. This software could be developed in-house, or an already-developed public-domain software could be used. In PASSER V, we decided to use the GA library (GAlib) developed by Matthew Wall (*13*) because of its flexibility and availability without cost and copyright regulations. GAlib allows selection of multiple GA parameters and operators for optimization, and it has capabilities that allow extension to user-defined operators. The library also allows extensive sensitivity analysis of GA parameters. In the future, these features will provide means for further research with several GA parameters not tested in this project. The version of GAlib selected for use in PASSER V is 2.45.

Advantages and Disadvantages of Genetic Algorithms

GAs have the following advantages over other optimization techniques:

- They do not require information about the nature of the optimization function.
- Discontinuities in the solution space have little effect on overall optimization performance.
- They are resistant to becoming trapped in local optima.
- They perform very well for large-scale optimization problems.
- They can be employed for a wide variety of optimization problems.

Disadvantages of GAs include difficulty in finding the global (best) solution, the need to evaluate a large number of possible solutions, and difficulties in implementation to a specific problem.

Terminology Applied in GAs

In the following subsections, we present terminology useful for the readers of this document (*13*).

<u>Types of Genetic Algorithms</u>

There are several types of GAs. In this project, we used the following two types:

1. simple genetic algorithm, and
2. steady-state genetic algorithm.

A simple genetic algorithm creates an initial population by cloning the individual or population passed when it is created. For each generation, the algorithm creates an entirely new population of individuals by selecting pairs of individuals from the previous population and mating them to produce two new offspring for the new population. This process continues until the stopping

criteria are met (determined by the terminator). A steady-state genetic algorithm applies overlapping populations with a user-specifiable amount of overlap. The algorithm creates a population of individuals by cloning the chromosomes of population that is passed when it is created. For each generation, the algorithm creates a temporary population of individuals, adds these to the previous population, and then removes the worst individuals in order to return the population to its original size. The amount of overlap between generations is selected by specifying the *Replacement* parameter. This is the percentage of the population that should be replaced each generation. Newly generated offspring are added to the population, then the worst individuals are destroyed (so the new offspring may or may not make it into the population, depending on whether they are better than the worst in the current population).

Elitism

Elitism applies only to a simple GA. Elitism means that the best individual from each generation is always carried over to the next generation.

Selection Scheme

The selection method determines how individuals are chosen for mating. If one uses a selection method that picks only the best individual, then the population will quickly converge to that individual. So the method should be biased toward better individuals; but it should also pick some offspring that are not quite as good overall but may have good characteristics. Some of the more common selection methods include: roulette wheel selection (the likelihood of picking an individual is proportional to the individual's fitness such as bandwidth or delay), tournament selection (a number of individuals are picked using roulette wheel selection, then the best of these are chosen for mating), and rank selection (pick the best individual every time).

Population Size

The size of the population in each generation quite often affects the solution. A population size of five to a population size of tens of thousands is used, depending on the evolutionary strategy and the nature of the problem that one is trying to solve. In a solution space of N possible solutions, a population of N individuals can solve the problem in 1 generation; however, N is often far too big (or unknown) to do that. Solution space affects the population size, hence multiple runs need to be conducted for each kind of problem to select the optimal population size.

Termination Criteria

GAs are terminated using two criteria: (1) convergence, and (2) number of generations. For this project, we defined convergence as the ratio of the average score of N previous best generations to the score of the current best-of-generation. One can also define the maximum number of generations after which the GA evolution should stop.

<u>Crossover Probability</u>

Crossover probability is the probability that two parents mate. An appropriate probability will allow parents to mate and thus search new solution spaces. In effect, evolutionary techniques are most useful for problems where the variables have complex, interacting dependencies, and a direct optimization algorithm is unknown. Selection and mutation alone cannot solve such problems when the solution space is large. Crossover is the real power behind evolutionary algorithms, and it improves performance by many orders of magnitude in most problems.

<u>Mutation Probability</u>

Mutation probability is the probability with which a given chromosome changes its state between generations. A high mutation probability will essentially lead to a random search of the solution space.

<u>Replacement Probability</u>

Replacement probability specifies the amount of overlap between generations. It only applies to steady-state GAs.

ALGORITHMS IN PASSER V

Several simulation or evaluation models have been included in PASSER V. These models are used by optimization algorithms and by other analysis tools. This section describes these models. A significant portion of this section is devoted to the new traffic model.

Volume Analysis Routine

This model was developed by Chaudhary et al. in a recent TxDOT project (*14*). For a given set of green splits for a facility (an arterial or interchange), this model calculates the maximum number of vehicles, per hour, that can go through the facility before some movement(s) becomes a bottleneck. This simple model does not consider the effects of blocking, and it is especially suitable for analyzing four-phase diamonds or facilities where sufficient storage space exists.

Bandwidth Analysis Routine

For a given timing plan (cycle length, splits, offsets, and phase sequences), bandwidth analysis routine (BAR) calculates the progression bands in both directions of an arterial. In its calculations, the routine calculates bands (a geometric quantity) between all signal pairs. This routine was developed for use in generating time-space diagrams and for use in bandwidth-based optimization using the genetic algorithm. After calculating the bands, this routine calculates bandwidth efficiency and attainability using Equations 5 and 6 provided earlier. This routine is extremely efficient in its calculations.

Delay Analysis Routine

PASSER V's delay analysis routine (DAR) employs mesoscopic simulation strategy. In other words, it simulates fractional flows and updates them every second. It performs the analysis of traffic conditions using a two-step process described below:

1. initialization, and
2. simulation and recording of MOEs.

For these steps, the model uses two subroutines: the undersaturated routine and the oversaturated routine. The program conducts the initialization step for two signal cycles. The first cycle uses the undersaturated routine to get preliminary estimate of queues, and the second cycle uses the oversaturated routine to ensure that the queue estimate is realistic. After the initialization step, the analysis step applies the oversaturated routine for a specified number of cycles. In PASSER V, this number is two cycles. DAR uses four types of movements: external-to-external, external-to-internal, internal-to-internal, and internal-to-external.

Undersaturated Routine

The routine assumes undersaturated flow conditions irrespective of the actual conditions in the network. This routine builds flows and queue profiles by applying an extended version of the delay-difference-of-offset (DDOF) model (*15*). This methodology is similar to TRANSYT 7F's link-wise simulation model. In the undersaturated step, the analysis is conducted one link at a time. Starting from the upstream link, each link is simulated. The upstream flow profiles are created and projected downstream. At the downstream intersection, the outflows and inflows are calculated and queue profiles and delay profiles are built. This process is repeated for each link. During this process, the routine applies the TRANSYT 7F platoon dispersion model (*4*). Queue storage on each link at the end of one cycle is obtained by building an input-output queue profile. If the queue at the end of the cycle is greater than the queue storage space, it is set equal to the storage space. The throughputs and delays for upstream and downstream movements are then calculated using flow profiles calculated using internal logic. The only exception is the external-to-external movements for which delay calculations use the Highway Capacity Manual (HCM) equation for calculating uniform and incremental delay (*16*).

Oversaturated Routine

The undersaturated routine treats a queue as an upward stack and, thus, is unable to model queue spillback and effects of upstream blocking as a result. In addition, it does not account for flow interactions between adjacent links. The oversaturated routine overcomes these limitations. It uses shockwave theory to more accurately assess delays in congested conditions. The program applies this routine during the second cycle of initialization and for all full simulation cycles. This routine conducts a second-by-second (stepwise) analysis of incoming flow at the stopbar, the available queue storage in the downstream link, and the outflow from the link. It updates conditions on all links of the arterial each second. The incoming flow at the stopbar for the internal movements is obtained by applying the TRANSYT 7F platoon dispersion model. The routine uses shockwave theory to keep track of the back of the moving queue at each link on a second-by-second basis. In addition, it keeps track of the available link storage. If the back-of-

queue reaches the upstream intersection, the available storage becomes 0 and movement blockage occurs until some storage becomes available. This routine performs calculations using the following steps:

1. For the first cycle (initialization period) of the oversaturated calculations, it obtains the queues stored at the end of link-wise simulation. If the queue is greater than the link storage space, it is adjusted to be equal to the link storage. DAR has the capability of keeping track of both movement-wise queue storage and lane-wise queue storage. For the initialization period, it uses the movement queue storage only.
2. Actual simulation starts using the flow profile from link-wise simulation together with the queue storage, movement storage, and lane storage from the previous step as the initial conditions.
3. It updates queue storage, movement storage, and lane storage for all links on a second-by-second basis. In the process, it applies platoon dispersion to the back of the queue and evaluates any link blockages and lane blockages.
4. It models link blockage and movement blockage, applying the second-by-second flows. This is performed by applying the following steps:
   - For each link, it first updates the downstream flows. For internal-to-external movements, the available movement storage and lane storage are reduced by the amount of outflow possible. For the downstream internal-to-internal movements, the outflow is updated considering the next link's available movement and lane storage.
   - For each direction, the internal-to-internal and external-to-internal movements for a given link are updated by obtaining the available movement/lane storage for the next downstream link. For calculating the available storage, shockwave theory is applied to find the actual available storage at each second.
   - If the next link is blocked, flows are stored in the current link itself.

The routine can perform step 4 for a specified number of signal cycles, and it calculates average MOEs over these cycles. Calculations performed by DAR use the following assumptions:

- Fractional flows may occur.
- Space inside an intersection never gets blocked.
- The only effect of queue spillback will be a decrease in flow from the upstream movements into this link.
- Lane blockages are only considered when all storage space of a lane is used.
- No right turns on red are allowed.

Lastly, DAR is limited to arterial systems. It is not capable of simulating traffic flow in multi-arterial networks. Furthermore, it is applicable to pretimed signals only. Kovvali provides a detailed description of DAR, along with its shockwave model, in his Ph.D. dissertation (*17*).

ANALYSIS OF OPTIMIZATION MODELS AND OPTIONS

This section provides results of research conducted to validate the analysis and optimization models in PASSER V.

Interference Minimization Algorithm

We conducted tests to compare the performance of this algorithm implementation against PASSER II. We found that, in most cases, the algorithm provided the same or better efficiency. In addition, this implementation selected smaller cycle lengths than PASSER II. In some cases, the bandwidth efficiency was less than that provided by PASSER II. However, these differences were minor. We conducted further studies to pinpoint the cause of this discrepancy. In these studies, we used cases with common cycle lengths and splits as input to both programs. These studies revealed that the cause of difference in bands was the differences in green splits calculated by the two programs. Further, we found that the algorithm implemented in PASSER V indeed performs better when we input the same splits in both programs.

Bandwidth Analysis Routine

We validated BAR by comparing the bands it produced against those produced by Synchro 4.0 and TSDWIN, version 2e (*18*). In this analysis, we used 20 different cases. Each case represented a given timing plan. For all these cases we found the directional bandwidths obtained from BAR, Synchro and TSDWIN to be within 2 seconds of each other. We also compared PASSER II output against Synchro and found that the differences in directional bandwidths between the two programs were within 2 seconds.

Delay Analysis Routine

Validation of DAR required a separate analysis of various features incorporated in the model. We used 20 replications of CORSIM simulations for comparing DAR with other models. From CORSIM output, we used delay and throughput for comparison purposes. In this stage, we considered the following two scenarios:

1. arterial system with spillback conditions due to short downstream link, and
2. arterial system with spillback conditions due to oversaturated downstream link.

Arterial System with Spillback Conditions Due to Short Downstream Link

For this analysis, we used a scenario used by Hadi and Wallace (*19*) for validating TRANSYT 7F, version 8.1. This scenario (Figure 5) consists of a one-way arterial link with two closely spaced intersections. The stopbar to stopbar length of the link is 325 feet. This link has two through lanes providing storage for 13.2 vehicles per lane, and it allowed investigation of conditions where the queue spillback occurs for undersaturated conditions due to bad offsets and short links with limited storage space. We selected a cycle length of 120 seconds to produce a volume-to-capacity (v/c) ratio of about 0.9 for both through approaches on the arterial. We studied two boundary conditions for the offset: (1) demand starvation, and (2) upstream traffic

movement blocking. Demand starvation is the condition where a movement has a green, but no output flow occurs because vehicles cannot get to the stopbar. DAR provides an estimation of demand starvation by recording the time during which the movement has a green but no output flow occurs due to zero inflows at the upstream intersection and no queue being in the link. Upstream movement blocking also decreases the throughput. Blocking can occur due to a bad relative offset or due to queue buildup. For this scenario, blocking occurred due to a bad offset. For an undersaturated link, the maximum downstream throughput can be achieved by eliminating starvation and blocking. Since the link for our test case is undersaturated, the maximum possible downstream throughput is equal to the traffic volume of 1400 vph used for this case. Similarly, the minimum arterial throughput occurs when the offset is such that only the vehicles stored on the link are output during the green.
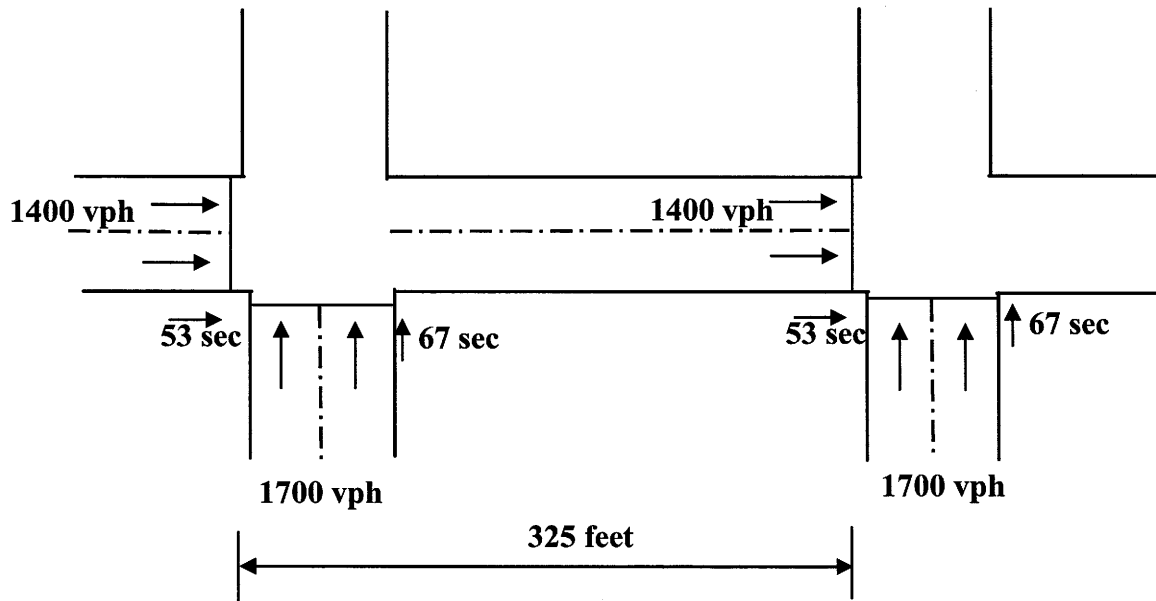
Figure 5. One-Way Link with Undersaturated Demand.

For this case, a total of 13.2 vehicles can be stored in each lane per cycle. Hence, the minimum throughput for a 120-second cycle length is 26.4 vehicles, which equals 793 vph. Figure 6 shows the relationship between offset and throughput for the arterial downstream movement as estimated by CORSIM and DAR for this scenario.

Figure 7 shows demand starvation time and downstream throughput for the arterial as offset varies between 0 and the cycle length. The figure shows that demand starvation leads to decreased throughput, as expected. Also note that throughput values estimated by DAR are similar to CORSIM results. For offsets between 40 and 80 seconds, the downstream throughput is equal to the vehicles stored in the link. CORSIM allows the storage of vehicles within the intersection area, whereas DAR does not. Because of this, CORSIM throughput is higher by 60 vph (2 vehicles per cycle). Figure 8 shows a comparison of the arterial delay estimated by DAR and CORSIM. The trends show that the decrease in throughput due to queue spillback is modeled accurately.

34
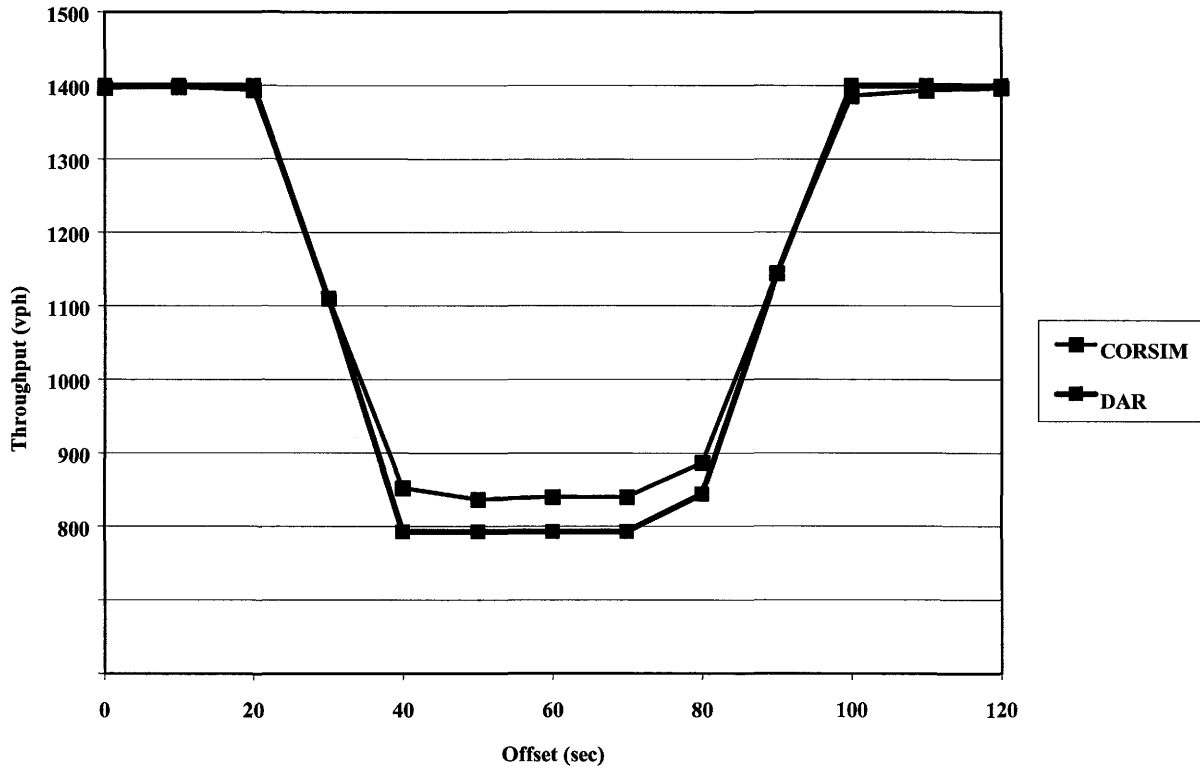
Figure 6. Downstream Arterial Throughput Estimated by CORSIM and DAR.
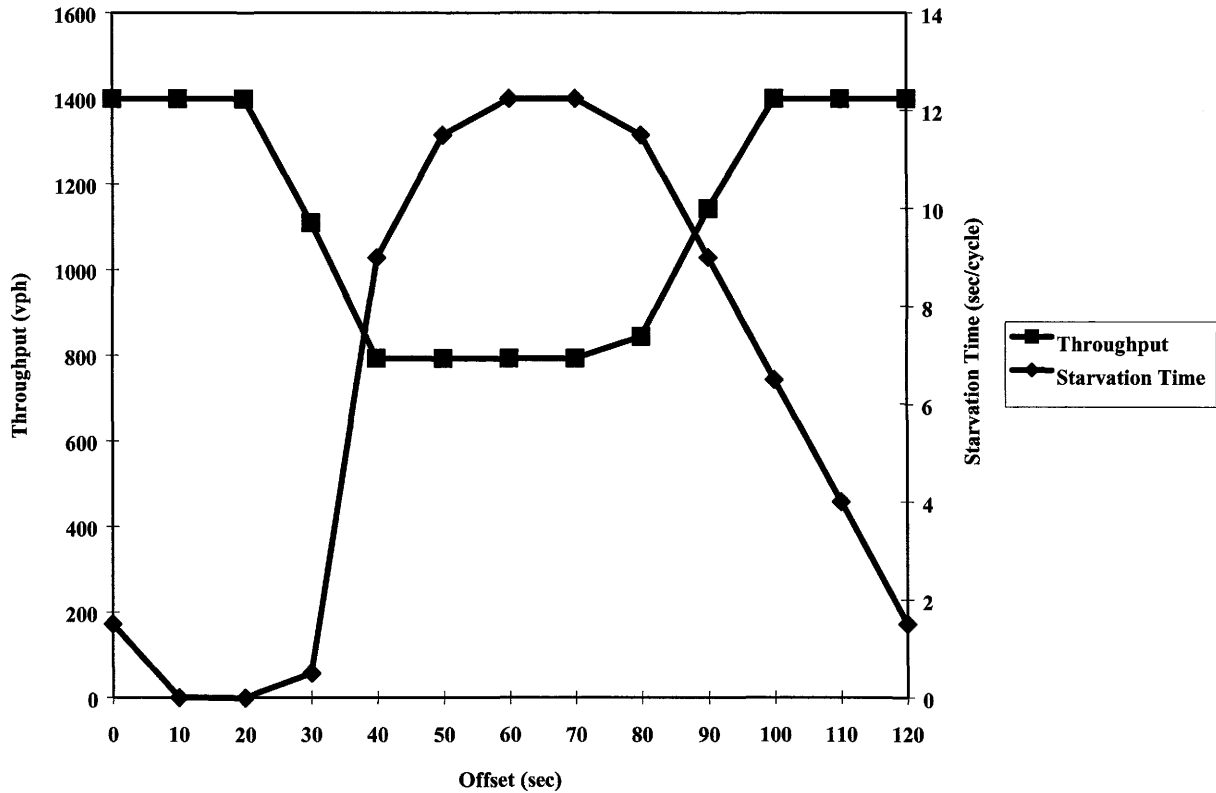
Figure 7. Relationship Between Demand Starvation and Throughput in DAR.
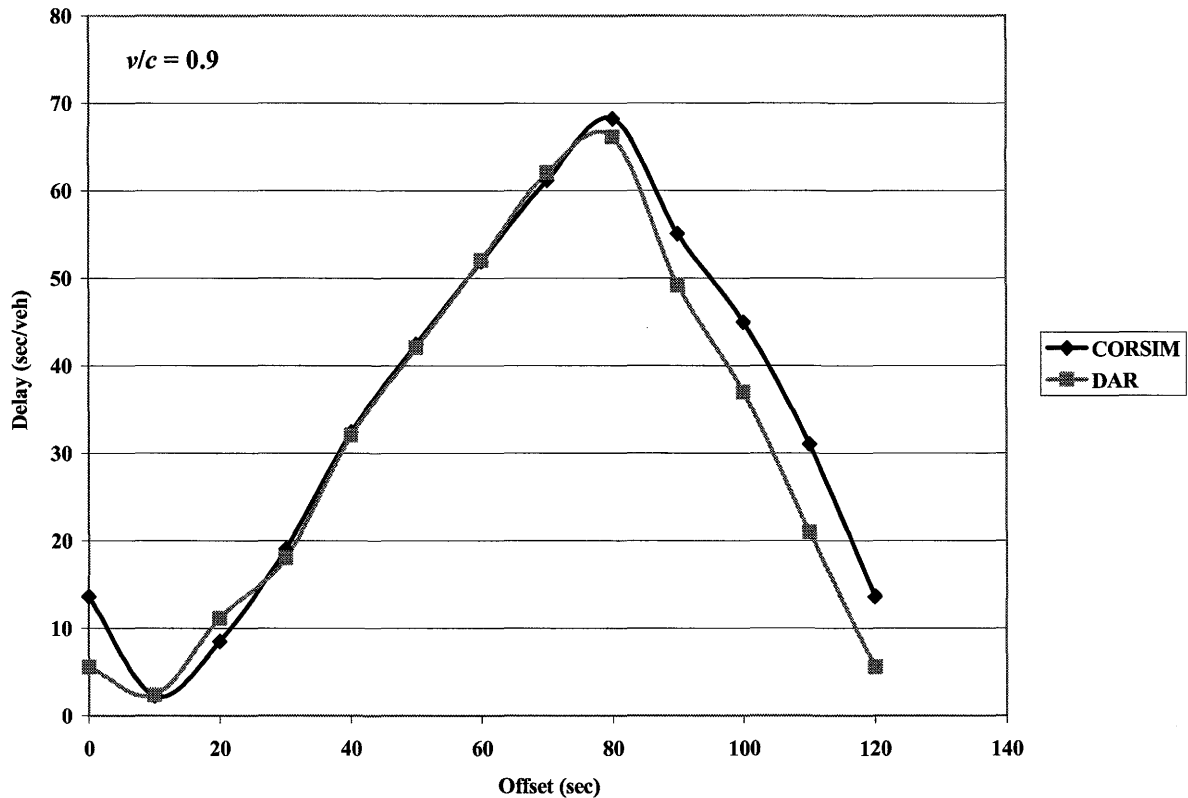
Figure 8. Comparison of Arterial Delay for CORSIM vs. DAR for a Short Link.

Arterial System with Spillback Conditions Due to Oversaturated Downstream Link

The simple arterial system used for investigating queue spillback due to an oversaturated downstream link is shown in Figure 9. We obtained this scenario from the paper published by Hadi et al. (*19*). For this scenario, the downstream arterial through movement capacity is less than the feeding volumes to this movement (v/c about 1.2). The distribution of the traffic to this movement is 75 percent arterial traffic and 25 percent ramp traffic.

We studied two boundary conditions for this scenario: (1) queue buildup and queue blocking, and (2) shockwave application for studying the movement of the back-of-queue. Because the downstream v/c ratio is only about 1.2, the queue buildup per cycle is low enough that the link is not blocked for several cycles of simulation in DAR. We simulated eight cycles (15 minutes) in DAR after the initialization period. From this analysis, we found that simulation needs to be conducted for three cycles for correctly modeling the back-of-queue for this case.

Figure 10 shows a comparison of the throughput from the CORSIM (NETSIM component) and DAR for various movements. We found some differences for offset values between of 60 and 70 seconds. Some of these differences are also unexpected because of the nature of the two simulation models. However, in general, we found that DAR results compared well with those in CORSIM.

36
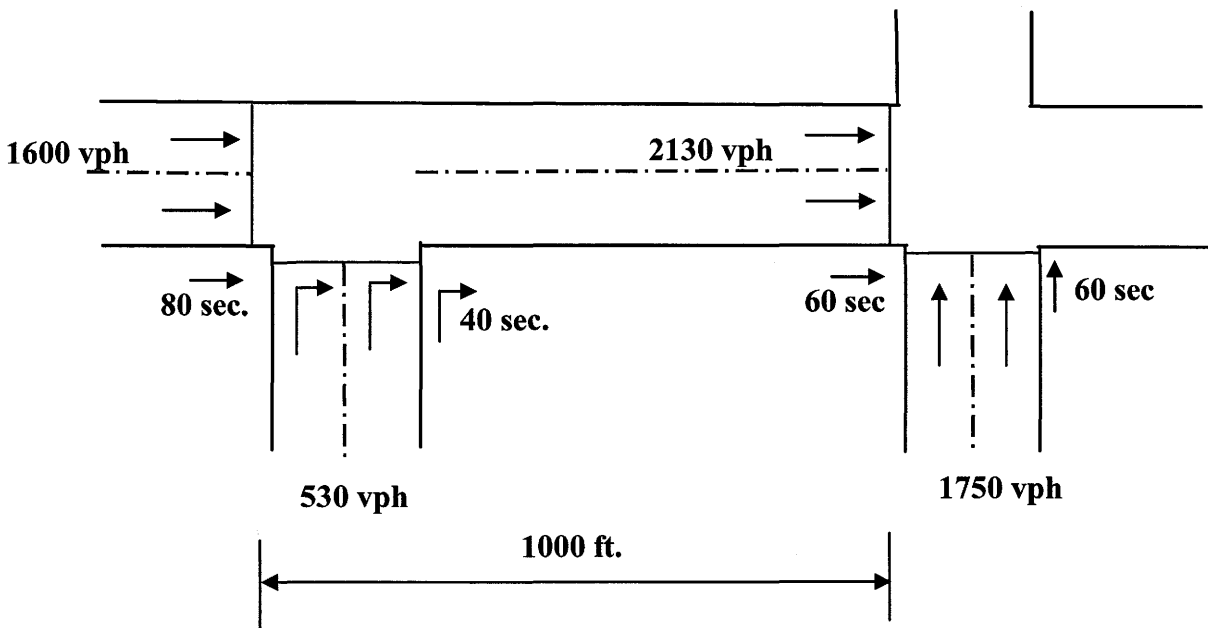
Figure 9. Scenario with Oversaturated Downstream Link.
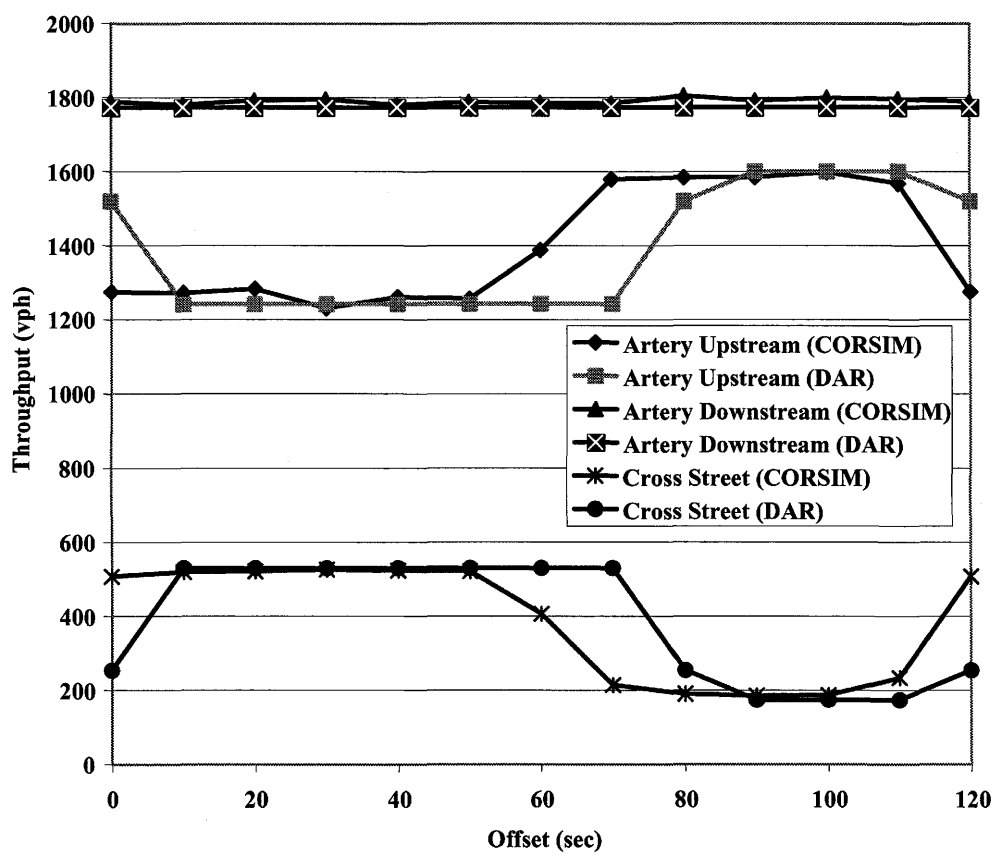
Figure 10. Throughput Comparison for CORSIM vs. DAR for Oversaturated Link.

Regardless of the offset chosen, the downstream through movement is always saturated for this case. Thus, the effect of offset is in determining the portion of the downstream link capacity assigned to the upstream arterial and cross-street traffic flows. Figure 11 shows the effects of spillback for the two upstream movements for the range of all possible offsets for this case. The results clearly illustrate that the offset dictates which upstream movement gets preferential treatment for such cases. This relationship observed between offset and flow is consistent with the results reported in a study conducted by Messer (20).
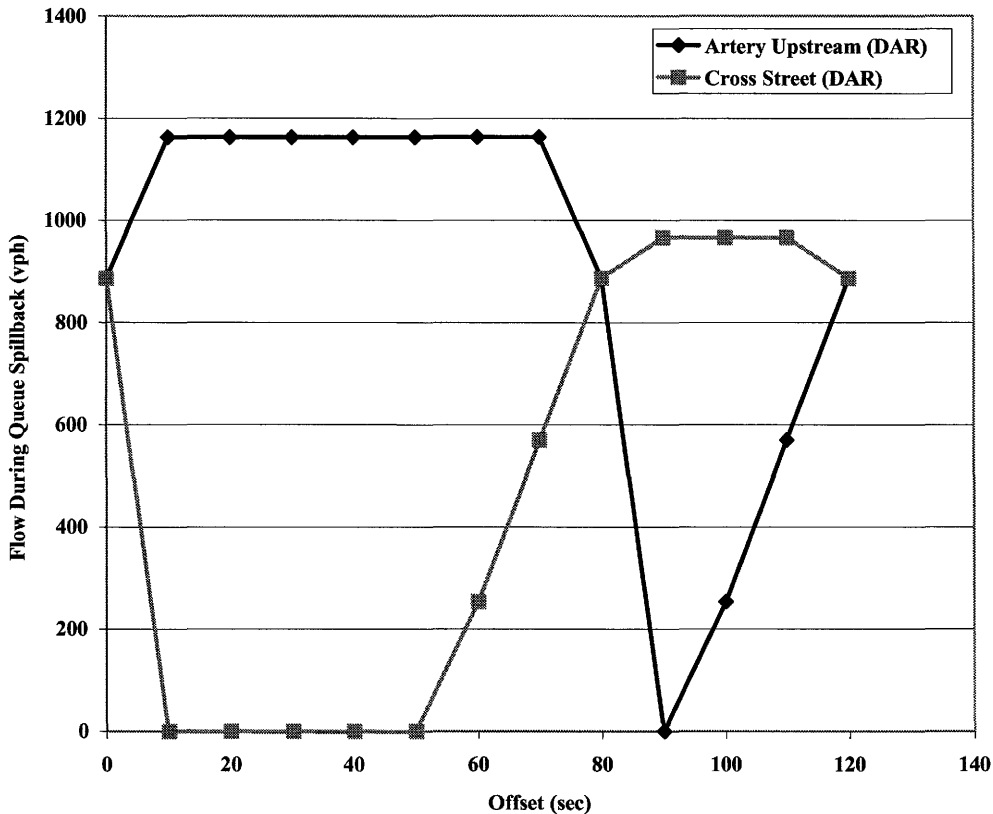
**Figure 11. Upstream Flow Impeded by Queue Spillback
for an Oversaturated Link in DAR.**

Figure 12 provides a plot of delays for the upstream and downstream through movements for the oversaturated link. As shown in the plot for the downstream through (internal) movement, the delay estimation in DAR closely follows CORSIM results. However, for the upstream external through movement, there are large differences. These differences are because of differences in the modeling approaches used by DAR and CORSIM. DAR does not consider storage requirements for external links; hence, all the demand to the movement during the time of study contributes to the delay. CORSIM, on the other hand, considers storage requirements on the external links, which means that only the actual volumes entering the link are delayed. These differences can be minimized by using large external links in CORSIM. The reader can see from Figure 12 that for the upstream external through movement DAR delays mostly follow the trends of CORSIM.
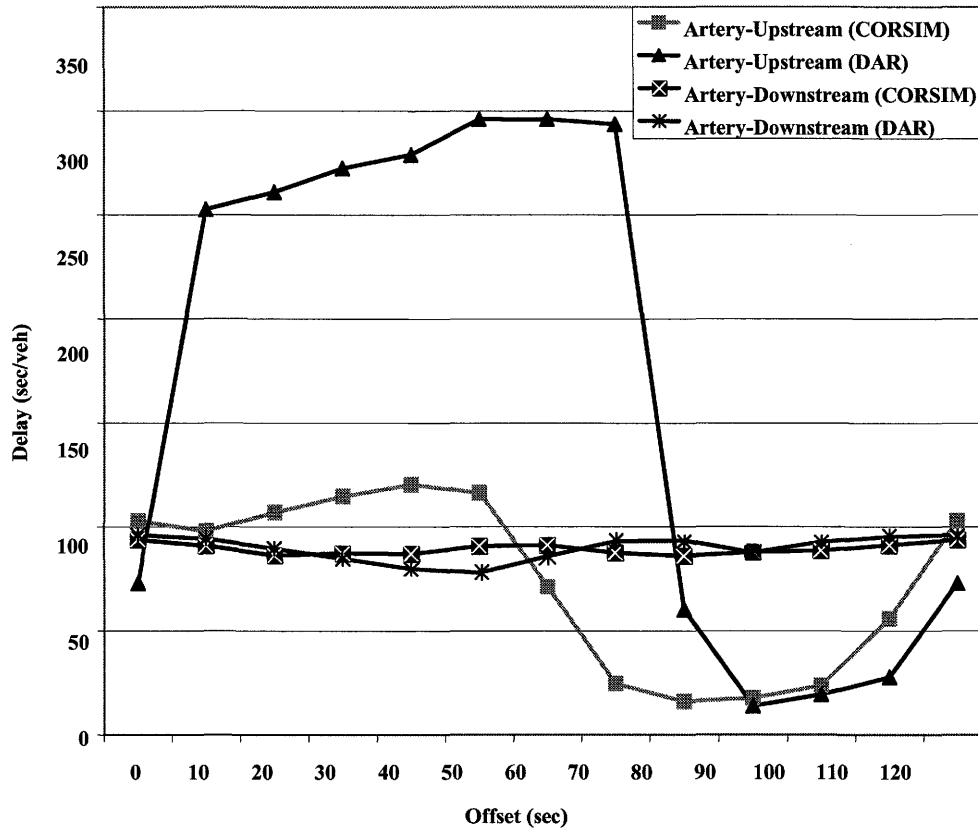
Figure 12. Delay Comparison of DAR and CORSIM for an Oversaturated Link.

Comparison of GA-DAR with PASSER III-99 and CORSIM

During the last step in the DAR validation process, we compared it with PASSER III and CORSIM for three different geometric conditions (interchange spacing of 200, 400, and 600 ft, respectively) and two different volume conditions (300 and 600 vehicles per hour per lane, vphpl). Figures 13 and 14 provide the results for Basic 3-phase and TTI 4-phase operations of the scenario with 200-ft spacing and 300 vphpl of volumes. The figures show that DAR results conform rather well to CORSIM and PASSER III results for both Basic 3-phase and TTI 4-phase timing plans.

Figures 15 and 16 provide the results of similar comparisons for 200-ft spacing and higher volumes of 600 vphpl. These volumes correspond to a volume-to-capacity ratio of about 0.9. The Basic 3-phase results for this condition indicate that PASSER III considerably overestimates interior delays. This overestimation is because PASSER III assumes vertical queue stacking and can store more vehicles in the link than is possible. The results obtained from DAR are more consistent with the CORSIM results. The variability in CORSIM and DAR is partly due to lane differences in the two models. We also found DAR results for the other volume and geometric conditions to be consistent with CORSIM output.
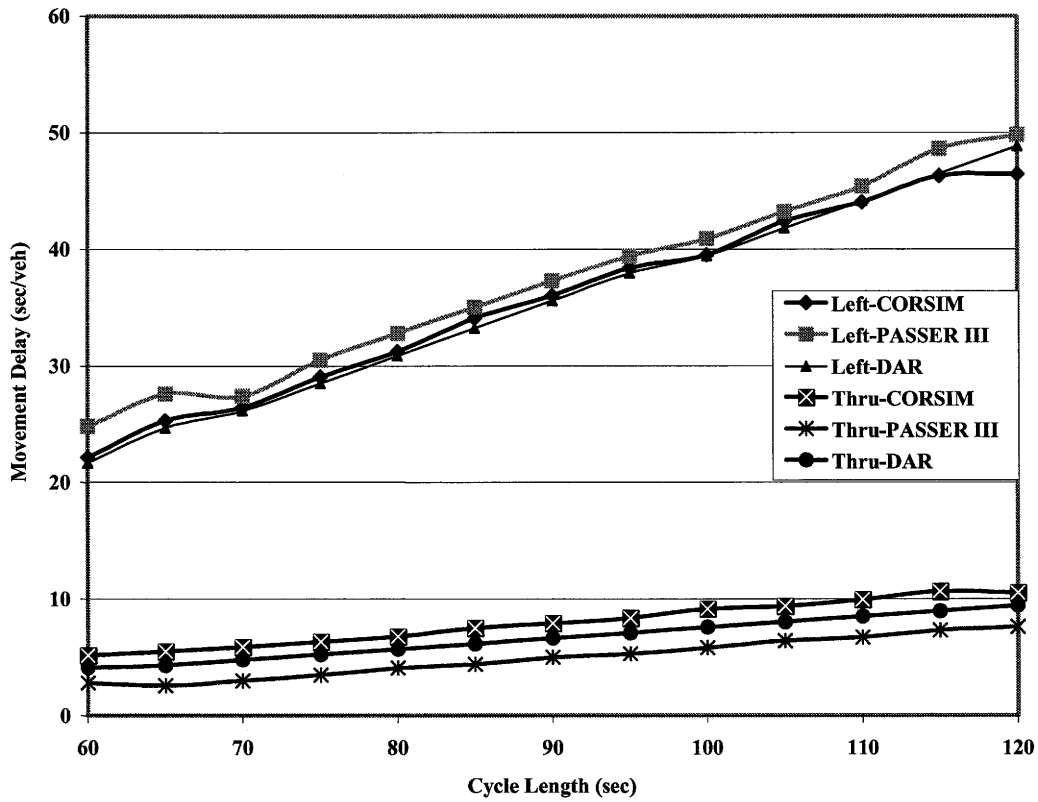
Figure 13. Validation of DAR for Basic 3-phase, 300 vphpl, and 200-ft Length.
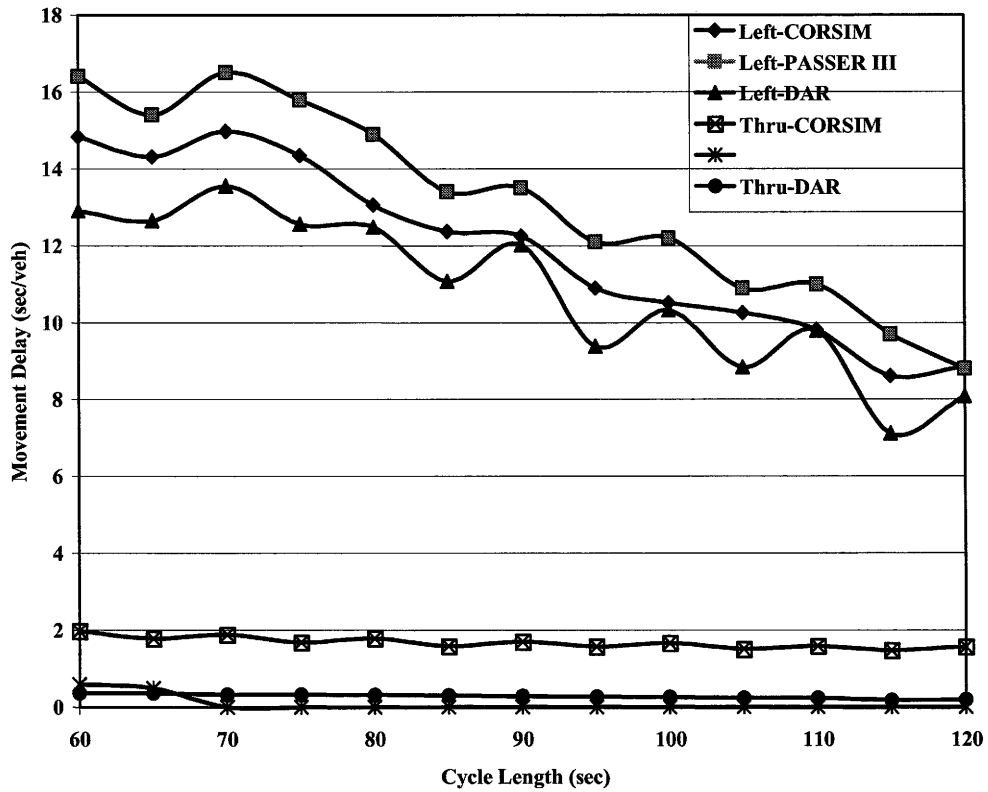
Figure 14. Validation of DAR for TTI 4-phase, 300 vphpl, and 200-ft Link.
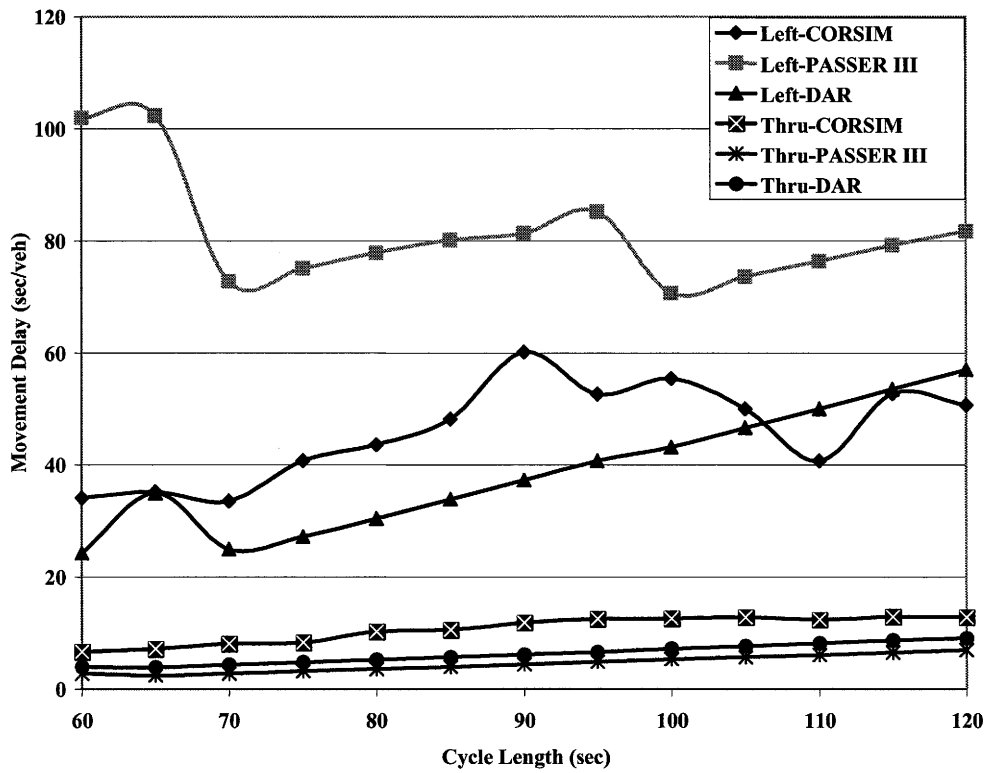
Figure 15. Validation of DAR for Basic 3-phase, 600 vphpl, and 200-ft Link.
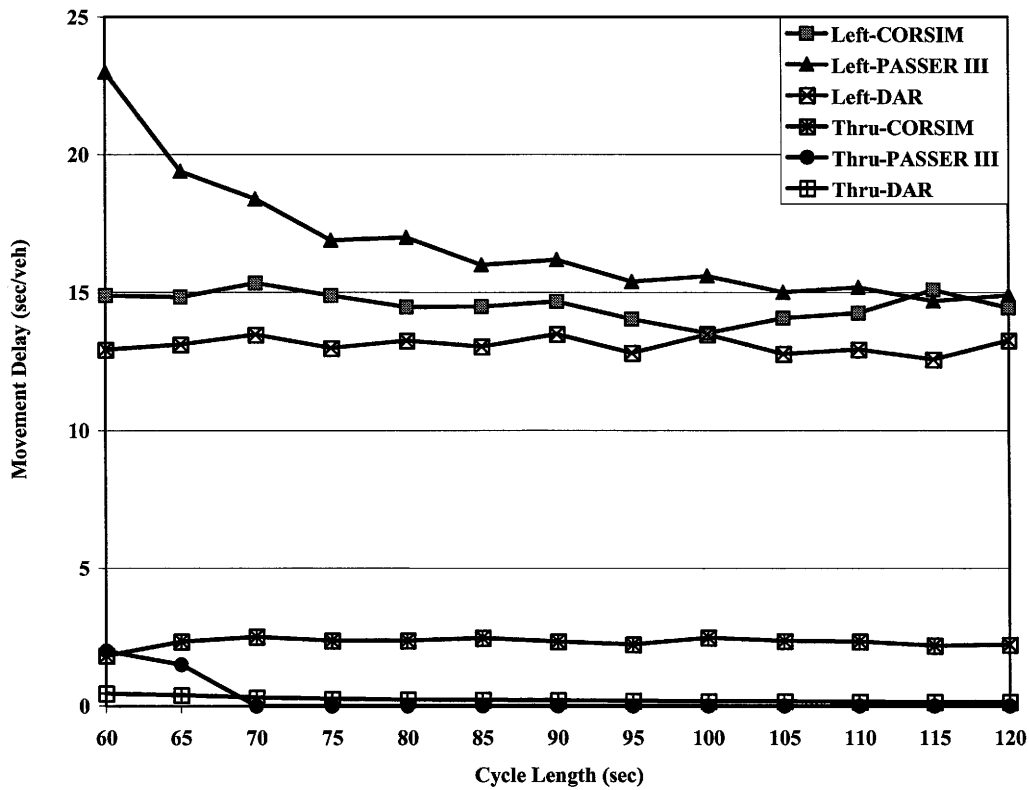
Figure 16. Validation of DAR for TTI 4-phase, 600 vphpl, and 200-ft Link.

Selection of GA Parameters

Selection of appropriate GA parameters can increase the computational efficiency and the possibility of obtaining the optimal result. This section provides a summary of studies conducted to study various parameters used by a GA during its search process. More technical and analysis details are provided in a Ph.D. dissertation by Kovvali (*17*). These studies were conducted for a four-intersection arterial illustrated in Figure 17. Only a subset of parameters was used. Table 9 identifies these parameters. These parameters create 140,000 analysis scenarios. Therefore, we employed other logical options to reduce the number of possibilities. Users interested in further detail should refer to the previous reference. Table 10 provides guidelines for selecting these parameters obtained from these studies.
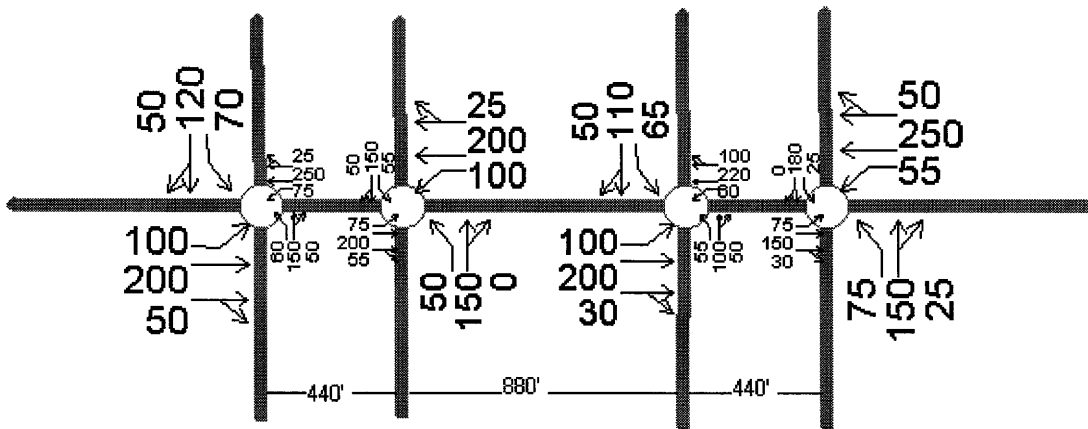
Figure 17. Arterial Used for the Study of GA Parameters.

Table 9. Subset of GA Parameters Studied.

Description	Variables	Cases
Type of GA	Simple vs. Steady-State	2
Type of Encoding	Binary vs. Real	2
Elitism (Simple GA only)	Yes vs. No	2
Selection Scheme	Roulette Wheel vs. Tournament Selection	2
Crossover Type	Single Point vs. Uniform	2
Population Size	10, 25, 50, 75 and 100	5
Number of Generations	50, 150, 250, 350 and 450	5
Crossover Probability	0.4 to 0.8; 0.1 increment	5
Mutation Probability	0.01 to 0.2; .01 increment	20
Replacement Probability (Steady-State GA Only)	0.4 to 0.8; 0.1 increment	5

Table 10. Guidelines for Selecting GA Parameters in PASSER V.

Description	Values
Genetic Algorithm	Simple Genetic Algorithm
Representation	Real Representation
Elitism (for Simple GA)	Yes
Selection Scheme	Tournament Selection
Crossover Type	Uniform
Population Size	50, 75, or 100
Number of Generations	350 or 450
Crossover Probability	0.5, 0.6, 0.7, or 0.8
Mutation Probability	0.05 to 0.19 in 0.01 increments

PASSER V provides features for selecting other options. Advanced-level researchers interested in additional analysis of GA parameters may do so using the program.

PASSER V USER INTERFACE

A new graphic user interface controls the operation of PASSER V. It provides access to data-entry functions and the optimization/analysis modules provided in the program. To use the program, a user will go through the following steps:

1. draw links to form a signalized arterial or a multi-arterial signal system,
2. enter link and node data,
3. define one or more subsystems,
4. use one of the tools to analyze or optimize the operation of a selected artery or subartery, and
5. view/print output for the selected facility.

Figure 18 provides an illustration of the main screen in PASSER V. This screen shows a six-intersection arterial defined in the currently opened data set. The program supports opening several data files simultaneously. In a data file, a user can define a network of arteries, however; the program allows optimization and analysis of timings for one arterial (or subarterial) at a time.

The main screen is divided into several subsections, including:

- menu options and toolbars with a variety of buttons,
- a map window, and
- a status bar.

Menu options and buttons provide functions for file management, data entry, and analysis and optimization using a variety of tools. The bar with large icons provides the most functionality needed for using the program. The main work palette is like a drawing board. It is the place where a user graphically defines the geometric structure of a signal system (i.e., an arterial).
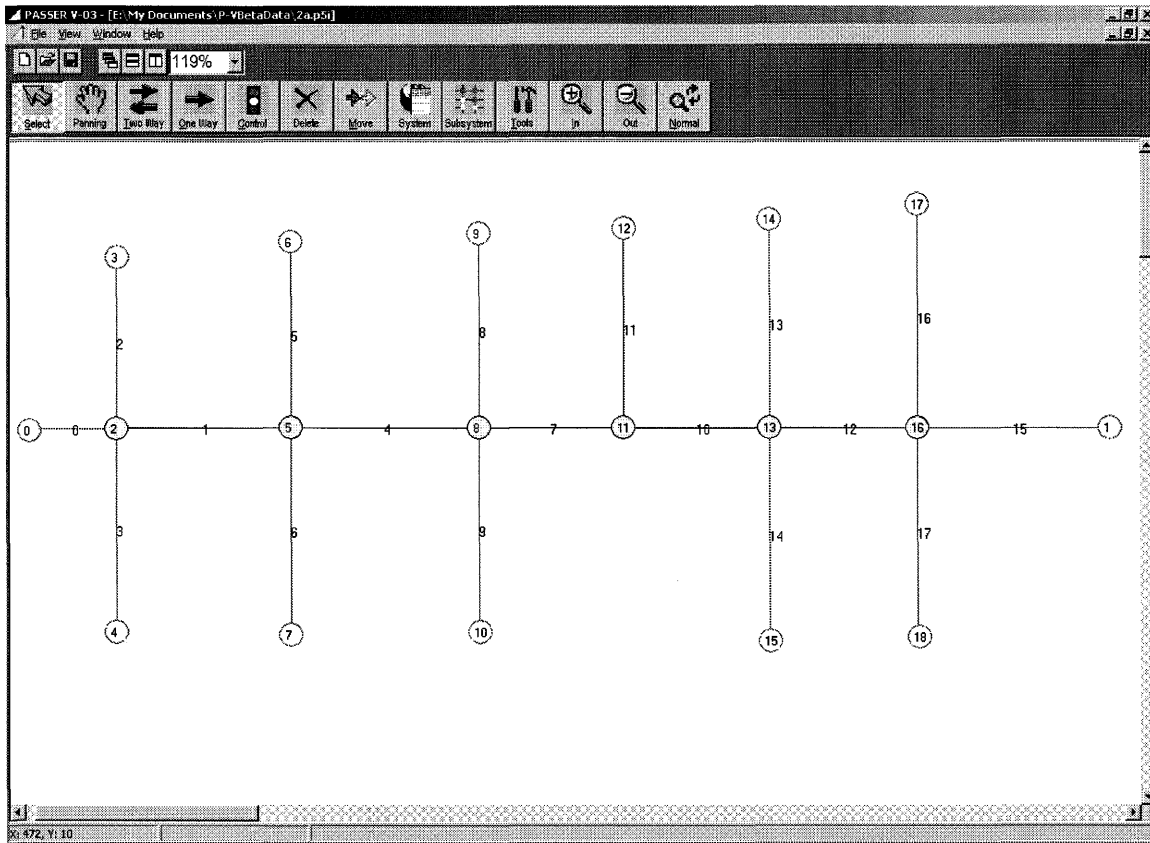
Figure 18. PASSER V Main Screen.

Input Module

The input module of PASSER V provides several features as described below.

Defining a Signal System

A signal system in PASSER V consists of a collection of links. To define a link, select a link by pressing the *Two-Way* or *One-Way* link button on the toolbar with large buttons. Then:

1. Place the crosshair in the map window, and press the left button on the mouse (left-button) to define one endpoint.
2. Move the mouse to another location in the window.
3. Press the left-button again.

This defines a link with two dummy signals as endpoints (nodes). Dummy nodes are identified by circles without any fill color. In PASSER V terminology, a node with two or fewer links is a dummy signal. The program automatically assigns numbers to links and nodes. To define a real signal, create another link crossing an existing link. The program identifies a node with three or more approach links as a real signal (circle with a fill). To define an artery, create links defining cross streets. Internally in PASSER V, a link is also identified as an artery. However, a real

44

artery must have at least two real signals. PASSER V calculates the length of each link drawn. Link distance is the distance between the center points of two nodes. The actual length of a link depends on the feet-per-pixel scale specified using the *System/Parameter* option. By default, the program assumes 5 feet per pixel. The user can request the program to display a grid to assist in drawing links with accurate lengths. Grid options are provided under the *View* submenu.

Entering Signal and Link Data

As mentioned above, the program assigns numbers to links and nodes. In addition, it assigns *x-y* coordinates to each node. The base reference point (0, 0) for these coordinates is the top-left corner of the map window. The user can click the left-button on the *Select* button (on the toolbar), and then click the left-button on a node or link to view or change these properties. The *Texas Diamond* selection on the Link Properties window can be used to identify a diamond interchange operated using a Texas diamond controller. Furthermore, if the user changes the *x-y coordinates* of a node, its physical location on the map also changes. The user can enter or modify all data for real signals by clicking the left-button on the *Control* option (icon with a signal head), followed by a mouse-click over a real signal. When done so, the program displays the screen shown in Figure 19. This screen contains several subsections that can be displayed by selecting tabs. In addition to entering data, the user can conduct isolated signal analysis here.
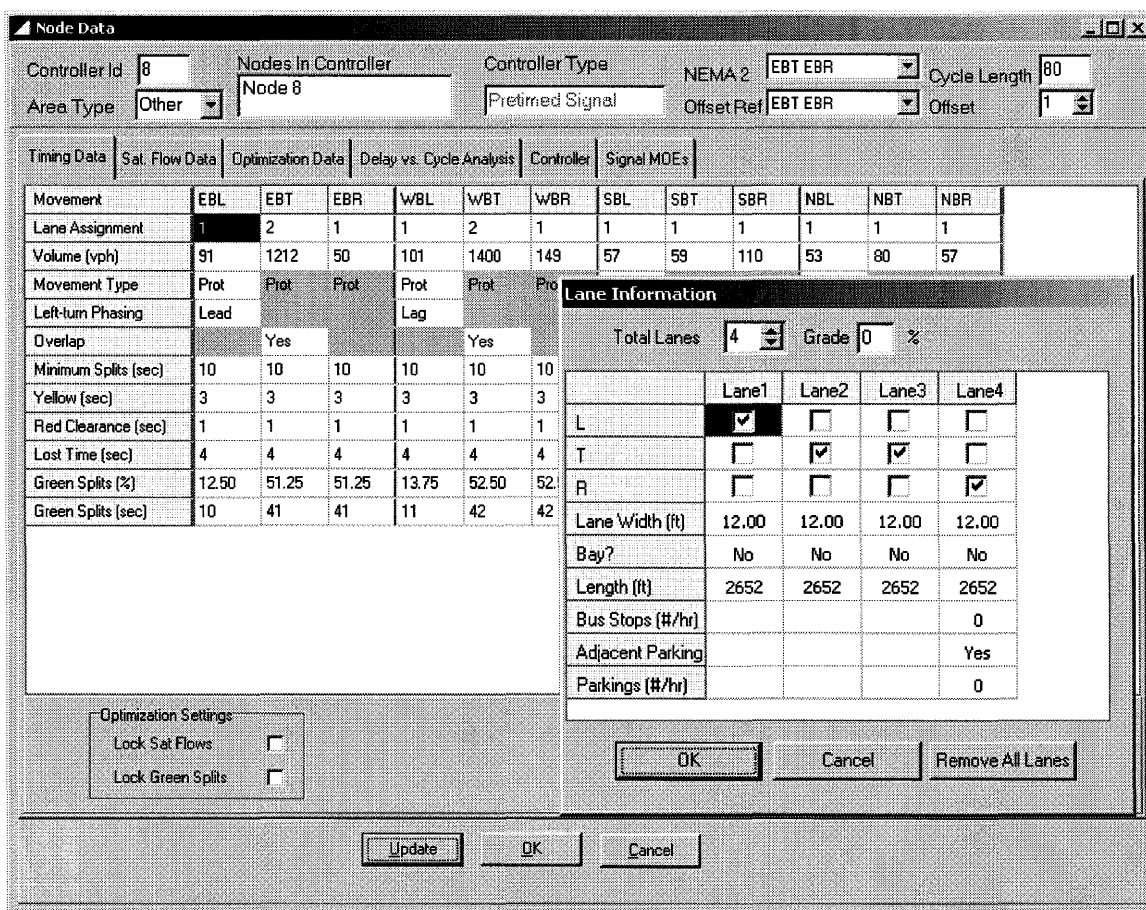
Figure 19. Signal Data and Analysis Screen.

45

After entering/modifying any data, including volumes and lane assignments, the user must click the *Update* button (on the bottom of screen) for the program to accept any modifications. At this point, the program executes two routines to calculate saturation flow rates and green-splits, and it fills the remaining data fields. The saturation-flow routine uses the same procedure used by PASSER III (*9*). This procedure calculates base flow rates for each lane and then prorates these rates using volumes and lane assignments entered by the user. The green-split routine calculates splits for regular signals by applying the Webster's model. It can also calculate splits for TTI 4-phase and Basic 3-phase options for Texas diamonds using a single controller. In this section, the program also provides MOEs for isolated signal timings selected. Lastly, the program provides features for performing HCM delay versus cycle-length analysis of an isolated signal as well. Figure 20 illustrates a sample output from such an analysis.
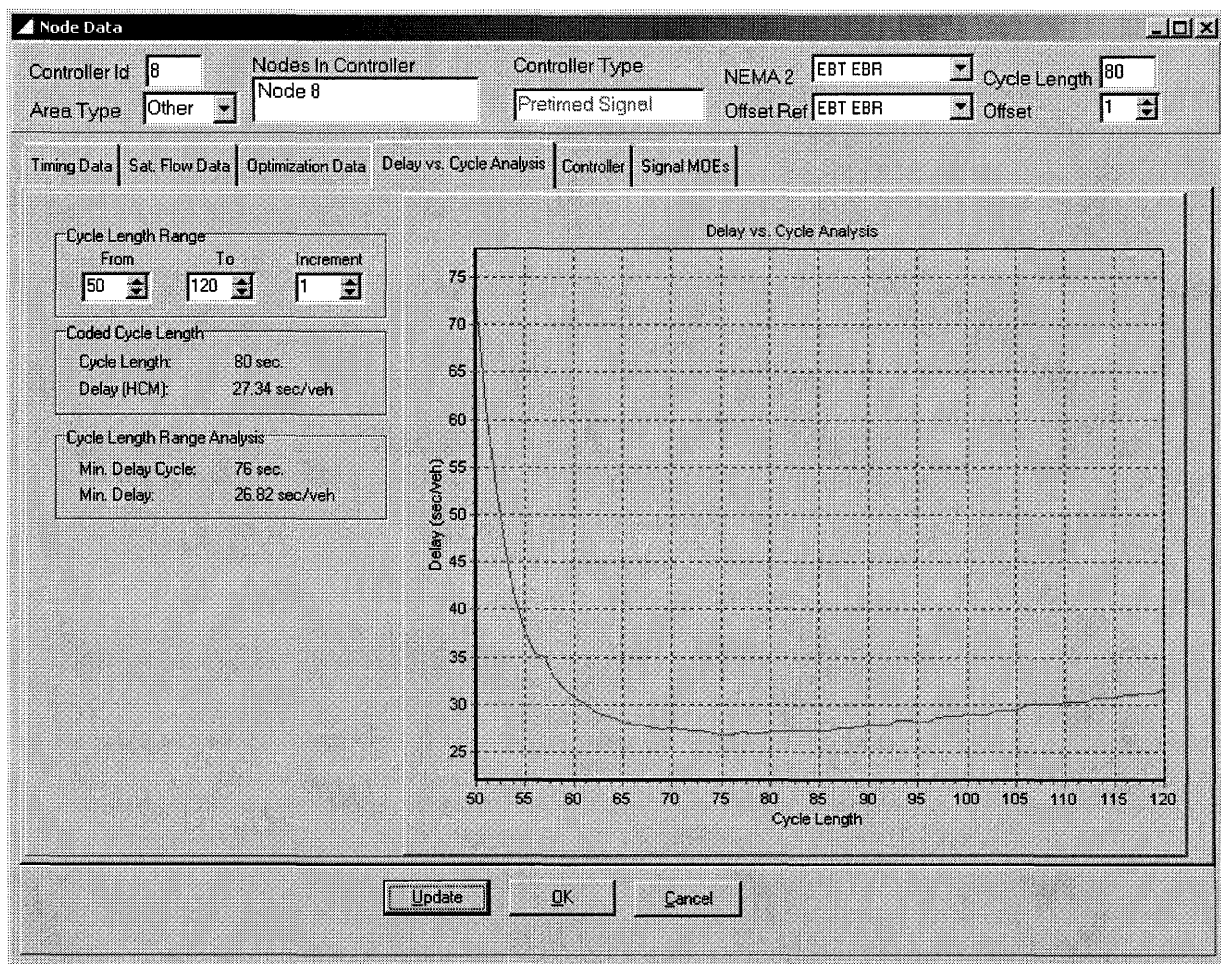
Figure 20. HCM Delay versus Cycle-Length Analysis in PASSER V.

46

Working with Signalized Arterials and Interchanges

As mentioned earlier, a user can create a network of signalized arterials; however, this version of PASSER V provides analysis and optimization functions for linear arterials only. All real arteries defined by the user are automatically identified by the program. In addition, the program provides a capability to define subarteries using the *Subsystem* button on the toolbar. A subartery consists of a subset of adjacent signals on an arterial. To define a subartery, enter a name for the subartery and click on the *Add* button. At this point the submenu disappears. On the map window, clicking the left-button on any number of adjacent links defines the subartery. The program automatically adds the subartery to the parent arterial. An example of a subartery is a diamond interchange. The user can select the *Subsystem->View* option to view a defined artery or subartery. At this point the user can use various applicable tools to analyze or optimize timings for any selected arterial. The user selects the *Tools* button to accomplish these tasks. When the user clicks the left-button on this icon, the program passes control to the Optimization/Analysis window. On the left side of this window, the program provides a list of all arterials defined by the user. The program displays a plus sign in front of an artery name if the user has defined subsystem(s) for that artery. The names of all subsystems for an artery can be displayed by clicking the left-button on the plus sign. As an alternative, the user can select the *Show All Sub-Arts* or *Hide All Sub-Arts* options to show or hide all sub-arterials defined. At this point, click the left-button on an artery or subartery to display all applicable tools (Figure 21).
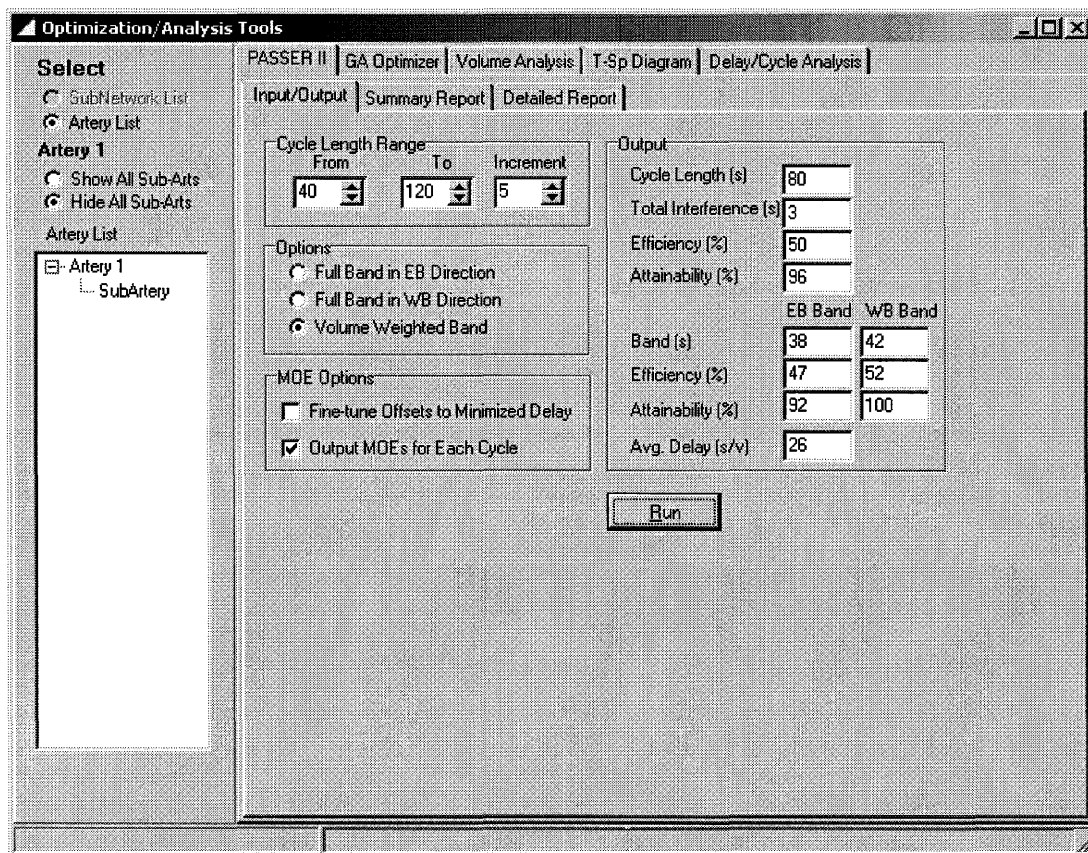
Figure 21. PASSER V Artery Analysis and Optimization Tools.

These tools include:

- *PASSER II* optimizes arterial signal timings for providing maximum arterial progression. This tool is only available when the arterial does not contain a Texas diamond interchange. This tool uses a re-engineered and enhanced version of the bandwidth optimization algorithm used by PASSER II.
- *PASSER III* optimizes the timings for a diamond-interchange using a Texas diamond controller. This tool is only available for an isolated diamond interchange and is similar to PASSER III program.
- *GA Optimizer* uses a genetic algorithm to provide for delay- and bandwidth-based signal timing optimization. The delay-based optimization feature is similar to features provided by Synchro and TRANSYT 7F. The bandwidth-based optimization is similar to the functionality provided by the *PASSER II* tool. This tool can be used to optimize timings for an arterial that contains Texas diamond interchanges.
- *Volume Analysis* tool provides a simple model for analyzing the throughput capacity of a system for specified timings. It identifies bottleneck locations in the system. This tool applies to all systems and provides accurate results only when blocking does not occur. An example of such a system is a diamond interchange with TTI four-phase operation. For such interchanges, this tool can also be used to analyze the effect of interior spacing on the interchange capacity.
- *T-SP Diagram* tool produces a graphic time-space diagram (illustrated in Figure 22).
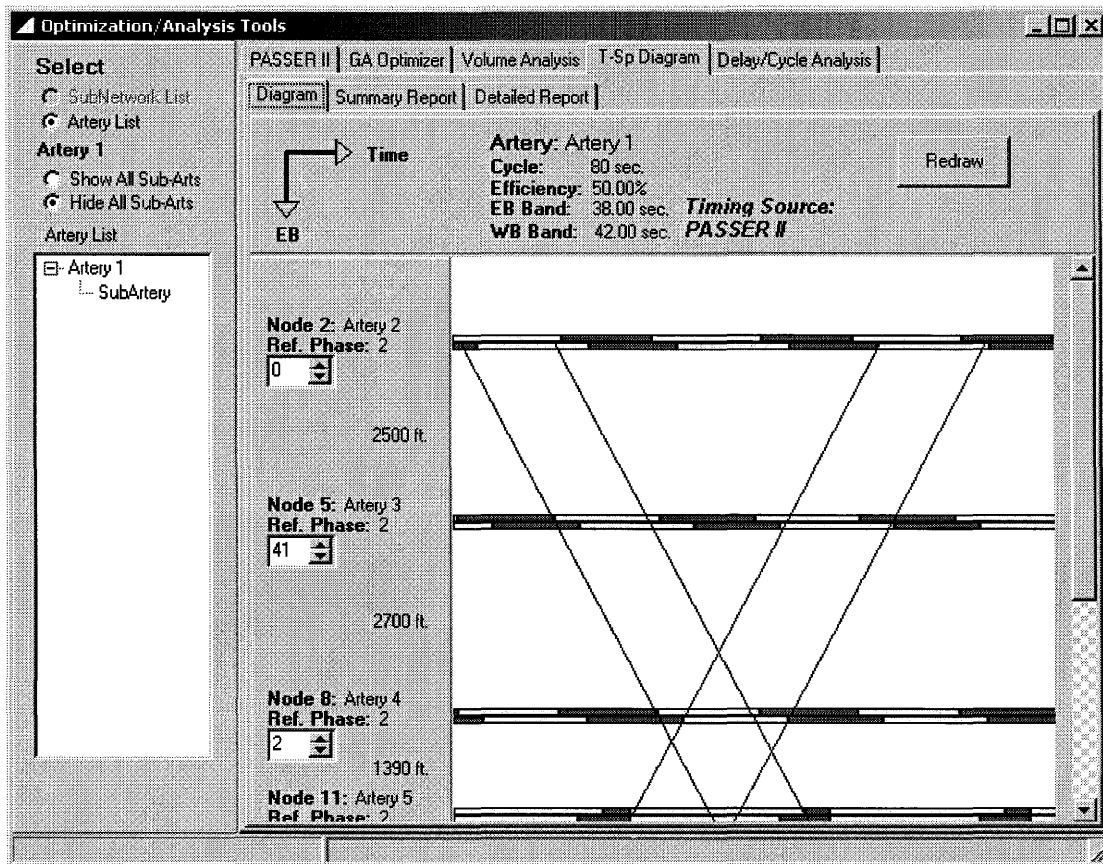
Figure 22. A Sample Time-Space Diagram.

48

- *Delay/Cycle Analysis* performs the delay analysis of a facility using PASSER V's delay analysis routine. It also provides a comparison of DAR results with delay computed using the HCM method. For the HCM method, the program computes delay for each signal in the system assuming random arrivals. Then, it computes the average delay for all signals. Figure 23 illustrates output from this tool.
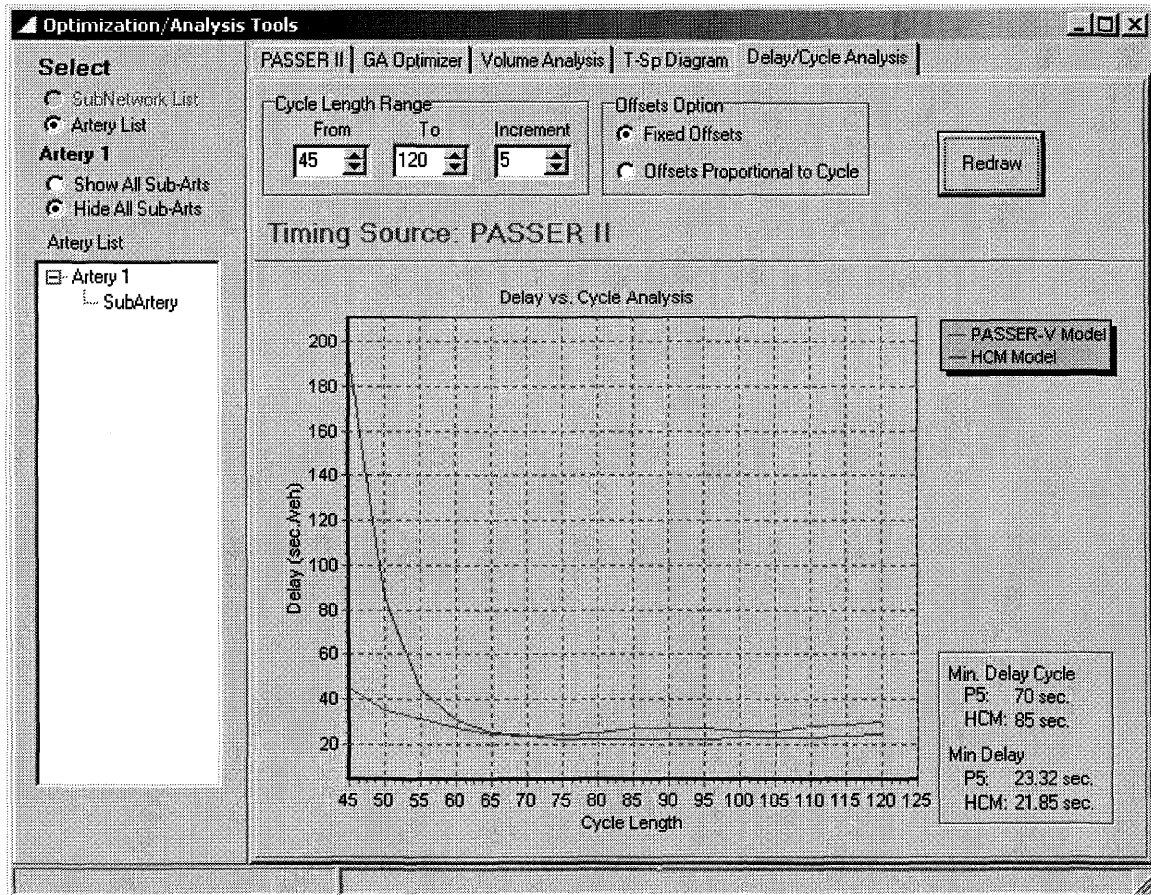
Figure 23. Delay versus Cycle-Length Analysis.

PASSER V displays a summary of results and detailed output for each timing plan analyzed. The number of timing plans available depends on the tool utilized. For instance, GA-based optimizer provides one best solution, whereas *PASSER II* and *PASSER III* tools can provide plans for each cycle length analyzed. Figures 24 and 25 provide samples of output screens. All program features are too numerous to describe in this report; however, descriptions provided in this section are sufficient for this report. In the next chapter, we provide the results of PASSER V comparisons with other popular programs.
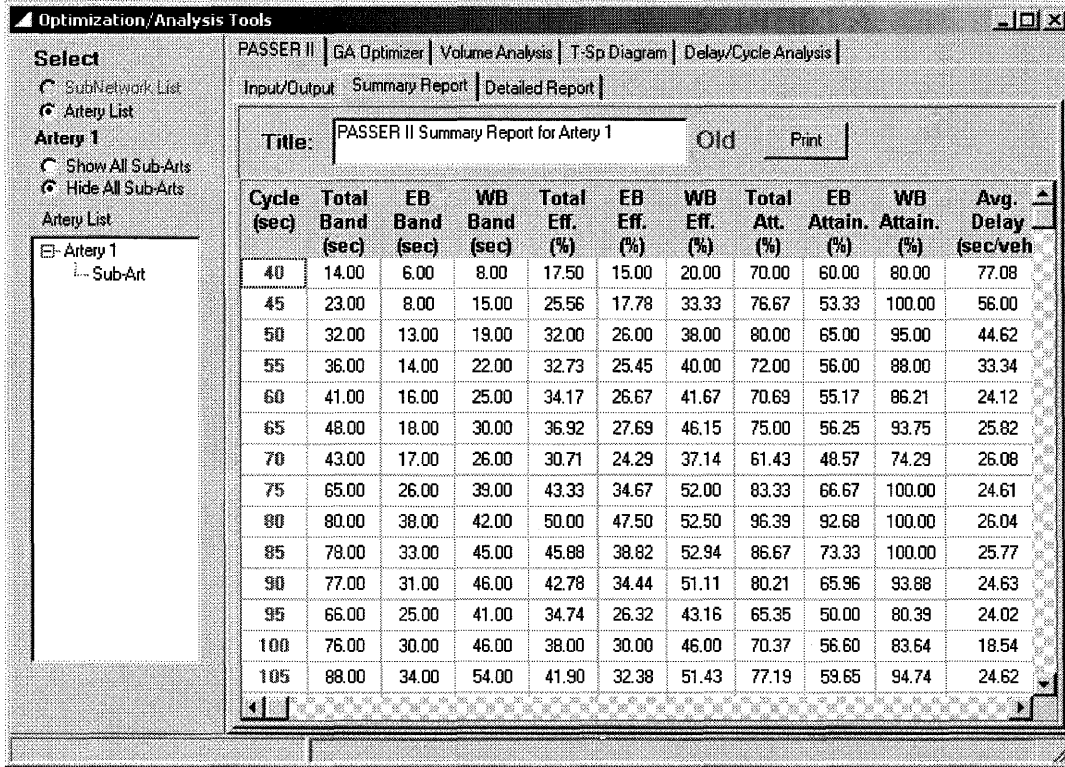
Figure 24. A Sample Summary Report from PASSER II Tool.
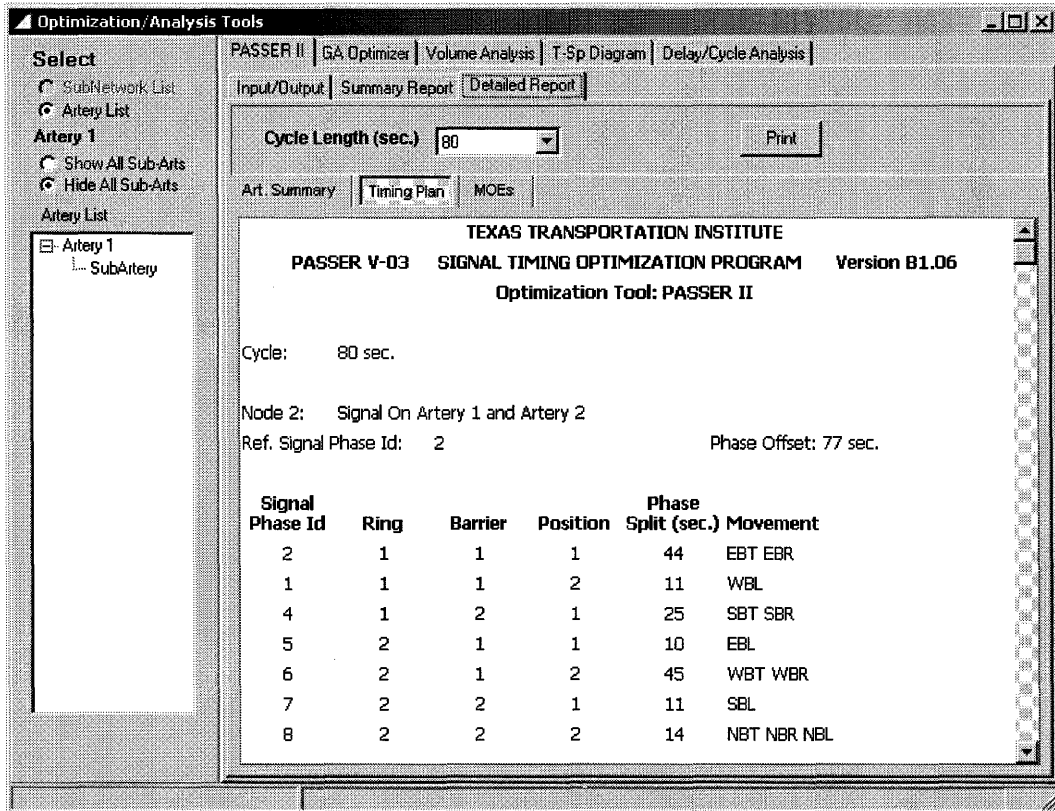
Figure 25. A Sample Signal Timing Report.

50

4. COMPARISON OF PASSER V WITH OTHER PROGRAMS

This chapter describes the results of studies conducted by the researchers to compare the performance of PASSER V with other software used previously. Specifically, we compared GA-based Bandwidth Analysis Routine (GA-BAR) and GA-based Delay Analysis Routine (GA-DAR) against PASSER II, PASSER III, TRANSYT 7F, and Synchro using CORSIM. We input optimal signal timings obtained from each program into CORSIM and conducted 20 replications of simulations using different random seed numbers. For each scenario, we averaged the results of CORSIM replications and compared the following two measures of effectiveness:

1. average system delay calculated by volume weighting delays for individual movements, and
2. system throughput, which is the sum of service volumes on all exit links.

OPTIMIZING DIAMOND INTERCHANGE TIMINGS

The green-split routine in PASSER V is capable of calculating special timings (TTI 4-phase, Basic 3-phase, and Extended 3-phase) for diamond interchanges used in Texas. In addition, the program also provides for analyzing standard phasing using two separate controllers. This latter case produces four possible phasing sequences, providing a capability of optimizing all possible diamond interchange timings. From a geometric point of view, a diamond interchange is a special case of an arterial with two closely spaced signals with one-way cross streets. The traffic flow patterns at diamond interchanges, however, are quite different from those at arterials. In general, these characteristics emphasize the need for using a delay-based model for analyzing and timing interchanges. Thus, we compared delay minimization using genetic algorithm (GA-DAR) against PASSER III (P3) for different volume and geometric conditions. The reader should note that PASSER III also uses a delay-based analysis and optimization model.

Scenarios and Search Options

The solution space for diamond interchanges consists of a combination of at most three variables: (1) cycle length, (2) phasing sequence, and (3) ring lag/internal offset. The possible phasing sequences are: TTI 4-phase, Basic 3-phase, Extended 3-phase, lead-lead, lead-lag, lag-lead, and lag-lag. The last variable for the GA optimization is the ring lag or internal offset. Ring lag applies only to standard phasing sequences (lead-lead, lead-lag, lag-lead, and lag-lag). Since the solution space is extremely small, we requested the GA to conduct analysis for a maximum of 20 generations using a population size of 20, a crossover probability of 0.5, and a mutation probability of 0.05. Also, we used *Simple GA* with real representation, elitism, uniform crossover, tournament selection, and no-scaling.

We compared the output MOEs from an average of 20 CORSIM replications for the low- and high-volume conditions for a 600-ft interchange spacing, and the high volume conditions for the 200-ft spacing. We conducted this analysis for four selected cycle lengths. We also requested the two programs to optimize timings over a range of cycle lengths (60 to 150 seconds) to select the best solution.

Summary of Results

For both volume conditions for the 600-ft spacing, no spillback occurred. However, for 200-ft spacing and high-volume conditions, spillback occurred. The results showed that the two programs selected similar timing plans when queue spillback was not an issue. We found no significant differences between PASSER III and GA-DAR for the low-volume condition. But for the high-volume and 200-ft spacing condition, GA-DAR outperformed PASSER III significantly. Figure 26 shows a comparison of the two programs for the high-volume and 200-ft spacing condition. As the reader can see, GA-DAR produced timing with higher system throughputs and significantly lower delays. This difference happens because PASSER III treats queues as an upward stack, while DAR models queue behavior as in reality. The vertical stacking of queues in PASSER III causes incorrect modeling of traffic, which leads to selection of the wrong optimal result as shown in the figure.
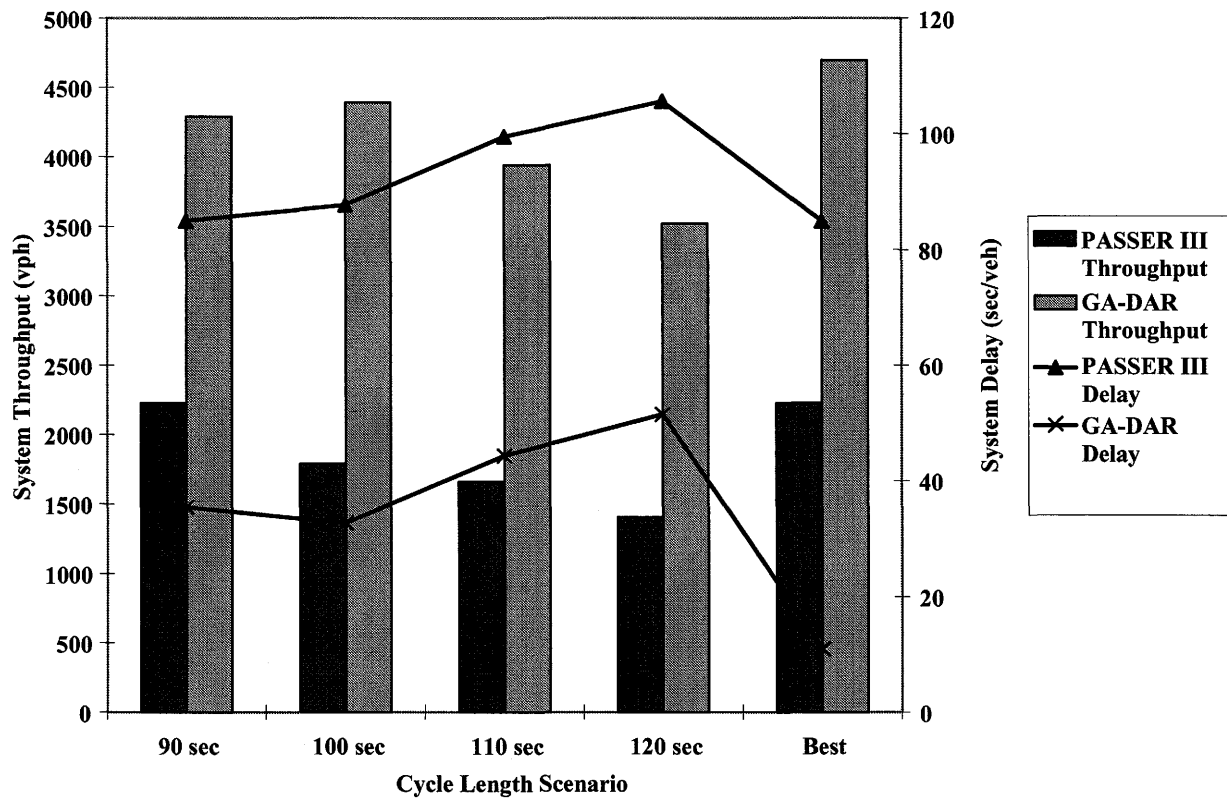
Figure 26. Comparison of Optimal Results from PASSER III and GA-DAR.

OPTIMIZING ARTERIAL SIGNAL TIMINGS

We tested the optimization performance of PASSER V's GA-based algorithms for arteries using the set of five small arterials described in Appendix A. These are the same arteries we used for comparing TRANSYT 7F, Synchro, and PASSER II earlier in the project.

Preliminary Studies to Select Green-Splits Calculation Strategy

Before performing a comparison of PASSER V with other programs, we conducted studies to analyze the performance of three green-split calculation strategies for GA-based optimization. These strategies are:

1. delay optimization with delay-minimized splits (DD),
2. delay optimization with Webster splits (DW), and
3. delay optimization with simultaneously optimized splits (DO).

In the first two strategies, GA uses predetermined splits to optimize offset, phase sequences, and offset. Strategy DD applies a GA-based delay minimization algorithm to precalculate splits. Strategy DW uses splits calculated using the standard Webster's method. The last strategy (DO) optimizes splits simultaneously with the other three signal-timing parameters.

PASSER V calculates least-delay splits used in DD by applying the GA-based optimization algorithm with the following fitness:

$$d_{splits} = \frac{\sum_{i=1 to n} d_i * v_i}{\sum_{i=1 to n} v_i} \qquad \text{Equation 7}$$

where:

d_i = HCM delay ($d_1 + d_2$) for movement i,
n = total number of movements at the signal, and
v_i = volume for movement i.

In concept, this method is similar to the strategy used in PASSER II for calculating minimum-delay splits. PASSER II, however, uses a hill-climbing optimization process.

Comparison of these three strategies for Arteries 1 and 2 first analyzed cases with fixed cycle length for each optimization run. These studies showed that there was no noticeable difference between Webster splits and delay-minimized splits, but simultaneously optimizing all four parameters resulted in much higher delays and lower system throughputs for the full-optimization case. The solution space for the full-optimization case is significantly larger than that for the cases where green splits are pre-calculated. Thus, the obvious reason for the poor performance might be that the evolution of the GA was not complete for the GA parameters selected. Additional studies are needed to investigate this issue. For the fixed-cycle cases for both arteries, we found simultaneous optimizations (DO) of all four parameters provided good results, but not much different from DD and DW. Hence, we concluded that optimizing green splits simultaneously does not provide any benefits.

Strategies Selected for Software Comparisons

Based on the preliminary analysis of green-split calculation strategies, we selected the two strategies that precalculate splits. These are: (1) delay-minimized splits, and (2) splits calculated using Webster's method. Also, we compared PASSER V's GA-based models for bandwidth-maximization (GA-BAR), and delay minimization (GA-DAR). Combined with the two green-split calculation options, these resulted in the following four strategies:

- delay minimization with delay-minimized splits (DD),
- delay minimization with Webster's splits (DW),
- bandwidth maximization with delay-minimized splits (BD), and
- bandwidth maximization using Webster's splits (BW).

In addition, we decided to use the GA parameters identified in Table 11.

Table 11. GA Parameters Selected for PASSER V Comparison Studies.

Description	GA-BAR Parameters	GA-DAR Parameters
Genetic Algorithm	Simple GA	Simple GA
Representation	Real Representation	Real Representation
Elitism (for Simple GA)	Yes	Yes
Selection Scheme	Tournament Selection	Tournament Selection
Crossover Type	Uniform Crossover	Uniform Crossover
Population Size	50	20
Number of Generations	350	150
Crossover Probability	0.7	0.5
Mutation Probability	0.08	0.05

Bandwidth Comparisons

During this stage, we compared the performance of GA-BAR optimization in PASSER V with that in PASSER II-90. For an unbiased comparison of these programs, we input the optimal timing plans from both programs into Synchro 4.0 and compared bandwidths calculated by Synchro. These comparisons were conducted in two stages described below.

Bandwidth Optimization for Selected Cycle Lengths and Splits

In this stage, we used Artery 1. Also, we used selected cycle lengths and splits calculated by PASSER II for each of these cycles. Both programs were asked to optimize offsets and phasing for given cycle lengths and corresponding splits. Table 12 shows the results of this comparison. The results indicate that the optimal values obtained from the GA model are very close to the PASSER II results for all four cycle lengths.

Table 12. Comparison of PASSER II and GA-BAR for Artery 1 Using Fixed Cycles and PASSER II Splits.

Cycle	Progression Efficiency (%)		Band A (Sec.)		Band B (Sec.)	
	GA-BAR	P2	GA-BAR	P2	GA-BAR	P2
90	27.22	27.22	28	29	21	20
100	26.5	27.0	31	32	22	22
110	26.82	27.72	35	36	24	25
120	26.67	27.08	37	38	27	27

Full Bandwidth Optimization

In this stage, we used all five arterials (Arteries 1-5, with base volumes) for comparing GA-based strategy in PASSER V against PASSER II. First, we optimized signal timings for Artery 1 using fixed cycle lengths of 90, 100, 110, and 120 seconds. The objective was to get a feel of difference in green-split calculations by the two programs. Then, we conducted optimization runs for all arteries over a cycle length range of 40 to 150 seconds with a cycle length increment of 5 seconds. Table 13 compares splits calculated by PASSER II and Strategy BW for Artery 1. We found that for most cases, PASSER II favored arterial through movements by calculating larger splits for through movements.

Table 13. Webster's Splits Calculated by PASSER V for Through Movements Compared to Corresponding Splits Calculated by PASSER II for Artery 1.

Signal	90		100		110		120	
	A	B	A	B	A	B	A	B
1	-1	0	-1	0	0	0	0	1
2	0	-1	-1	-2	-2	-2	-4	-4
3	-4	-3	-4	-3	-6	-5	-3	-3
4	0	-1	1	1	0	1	2	2
5	-4	-3	-7	-5	-5	-4	-8	-7
6	-2	-2	-3	-3	-4	-5	-6	-6

*Negative numbers indicate PASSER V splits are less than PASSER II.

Table 14 provides a summary of bandwidth efficiency produced by PASSER II, BW, and BD. For three cases, PASSER II selected significantly smaller cycle lengths than the other two strategies, and for one case it selected a much larger cycle length. However, a comparison of bandwidth efficiencies shows no significant difference in the performance of the two programs. Table 15 provides band attainabilities for the same scenarios. Again, both programs gave comparable results. There were some differences, but we found no trends in the differences. We submit that the differences are due to difference in green splits calculated by the two programs.

Table 14. Comparison of PASSER II and PASSER V Bands when Cycle Lengths Optimized.

Artery	Progression Efficiency (%)			Cycle Selected		
	P2	BW	BD	P2	BW	BD
1	26.67	27.50	26.80	90	120	125
2	48.75	43.81	43.64	80	105	110
3	26.67	28.00	30.50	135	100	100
4	26.36	28.82	31.38	55	85	145
5	23.33	24.78	26.67	120	115	135

Table 15. Comparison of Attainability for PASSER II and PASSER V Bands.

Artery	Total Attainability (%)			Attainability A (%)			Attainability B (%)		
	P2	BW	BD	P2	BW	BD	P2	BW	BD
1	96.55	97.06	97.10	95.24	95.00	95.12	96.00	100	100
2	94.87	80.00	74.42	100	59.65	68.18	97.50	100	80.95
3	97.67	100	100	100	100	100	98.63	100	100
4	100.00	100	98.91	61.11	100	100	80.56	100	96.55
5	96.88	100	100	100	100	100	98.25	100	100

We used a Pentium III 450 MHz computer with 256 MB RAM for these runs. PASSER II run times varied between 1 and 3 seconds, while the GA-based bandwidth routine took between 30 seconds and 45 seconds.

Delay and Throughput Comparisons

Versions of Synchro 4, PASSER II-90, and TRANSYT 7F programs used at this stage were the same as those used for the initial stages of software comparisons. The following identifies the version of each program and its features utilized in this project:

- TRANSYT 7F, version 8.2: Step-wise model was applied for the analysis. The optimization runs consisted of a 5-minute initialization period followed by a 15-minute analysis period. Since TRANSYT 7F cannot optimize phase sequences, leading left-turn phases were selected for all TRANSYT 7F runs.
- Synchro, version 4.0, Build 223: In Synchro, the percentile delay method and extensive optimization were selected. In addition, uncoordinated signals and double-cycled signals in the system were not allowed for the runs conducted for this research.
- PASSER II-90, version 2.0: PASSER II was allowed to optimize all signal timing parameters.

We compared three software packages and the four PASSER V strategies (DD, DW, BD, and BW) in this stage. The delay and throughput comparisons for arterials were conducted in two stages presented in the following subsections.

Optimization for Selected Cycle Lengths

During this stage, we compared PASSER V, PASSER II, TRANSYT 7F, and Synchro for selected cycle lengths of 90, 100, 110, and 120 seconds using Artery 1. We used the *Maximin* criterion for selecting these cycle lengths. The objective of comparisons in this stage was to study the effects of offset, phasing sequence, and green splits on optimization.

Figures 27 and 28 provide delay and throughput comparisons of each of the cycle lengths for Artery 1. We made the following conclusions from the graphs and additional statistical analysis:

- TRANSYT 7F version 8.2 is significantly and consistently worse than the other six strategies studied for both delay and throughput comparisons.
- The four PASSER V strategies and PASSER II are significantly better than Synchro for both delay and throughput comparisons.
- All four PASSER V strategies were significantly better than PASSER II for delays, but only the DW and BD strategies were significantly better than PASSER II for throughput comparison.
- BD strategy was significantly better than BW for throughput comparison, but not for delay comparison. No significant differences were found between the DD and DW strategies.
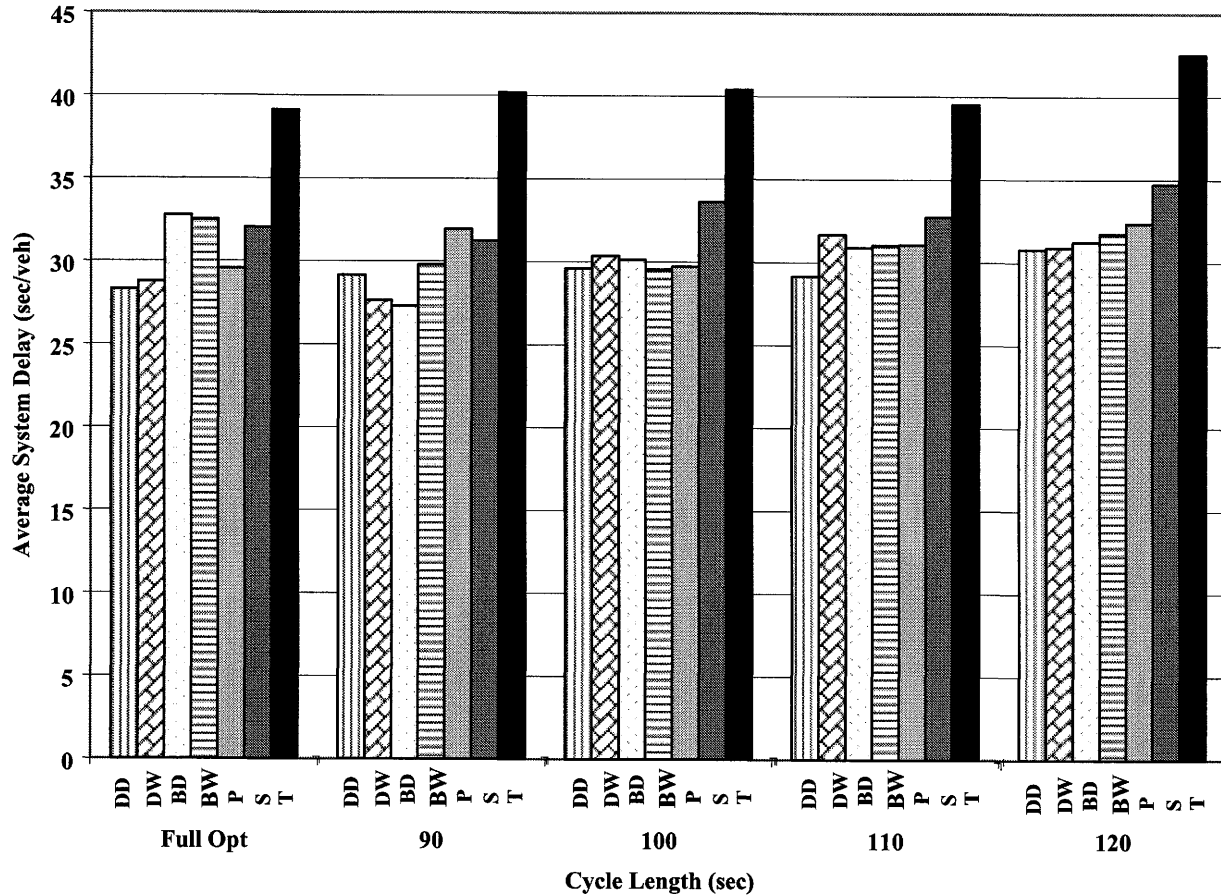
Figure 27. Delay Comparison for Selected Cycles for Artery 1.

57
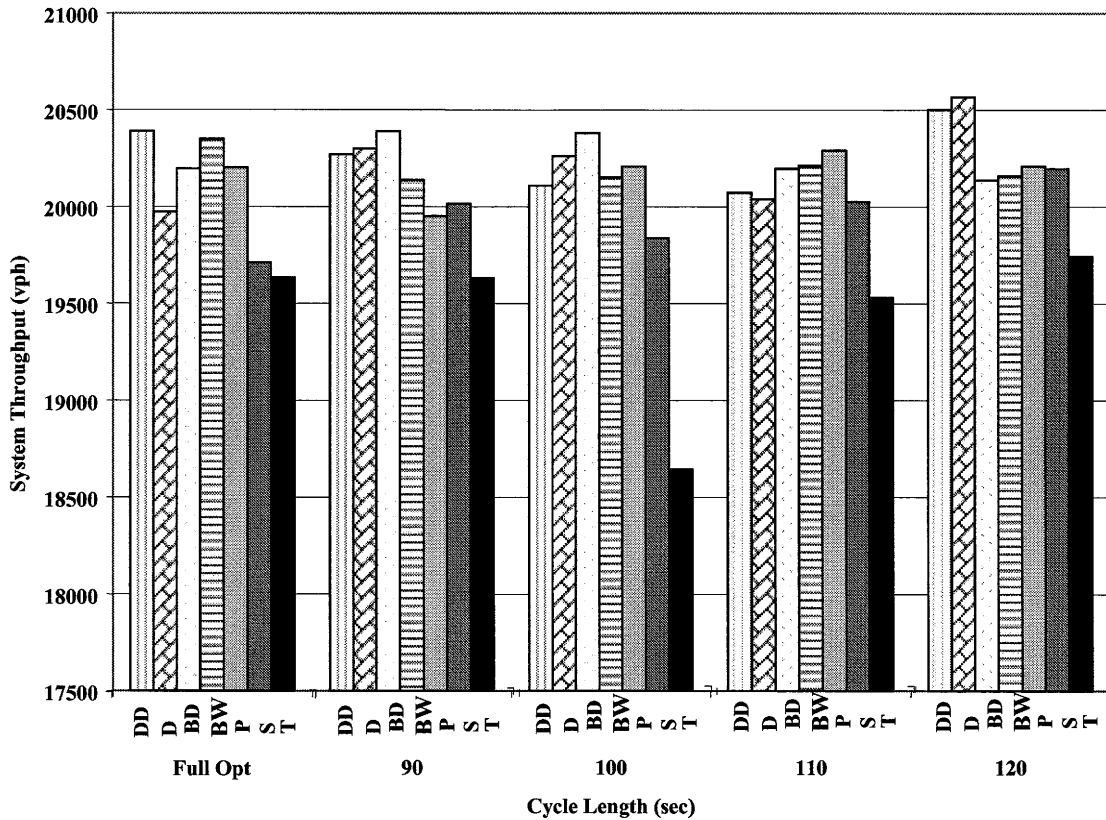
Figure 28. Throughput Comparison for Selected Cycles for Artery 1.

Overall, the four PASSER V strategies consistently provide good results in this stage of analysis.

Full Optimization

In this stage, we used all five arteries and optimized all four signal timing parameters over a cycle length range of 40 to 150 seconds. This stage represents the kind of analysis a traffic engineer would likely perform, where the objective is to find the optimal result possible without placing any significant constraints on the optimization. Table 16 shows the optimal cycle lengths selected by the programs. We found that delay-based GA optimization routines selected cycle lengths lower than the bandwidth-based GA optimization routines. These results are consistent with expected results. Figures 29 through 38 provide plots of delay and throughput.

Table 16. Optimal Cycle Lengths Selected for the Five Arteries.

Strategy	Artery 1	Artery 2	Artery 3	Artery 4	Artery 5
DD	100	60	70	65	95
DW	75	65	80	65	100
BD	125	110	100	145	135
BW	120	105	100	85	115
P	90	80	135	55	120
S	70	80	75	85	100
T	120	95	45	60	100

58
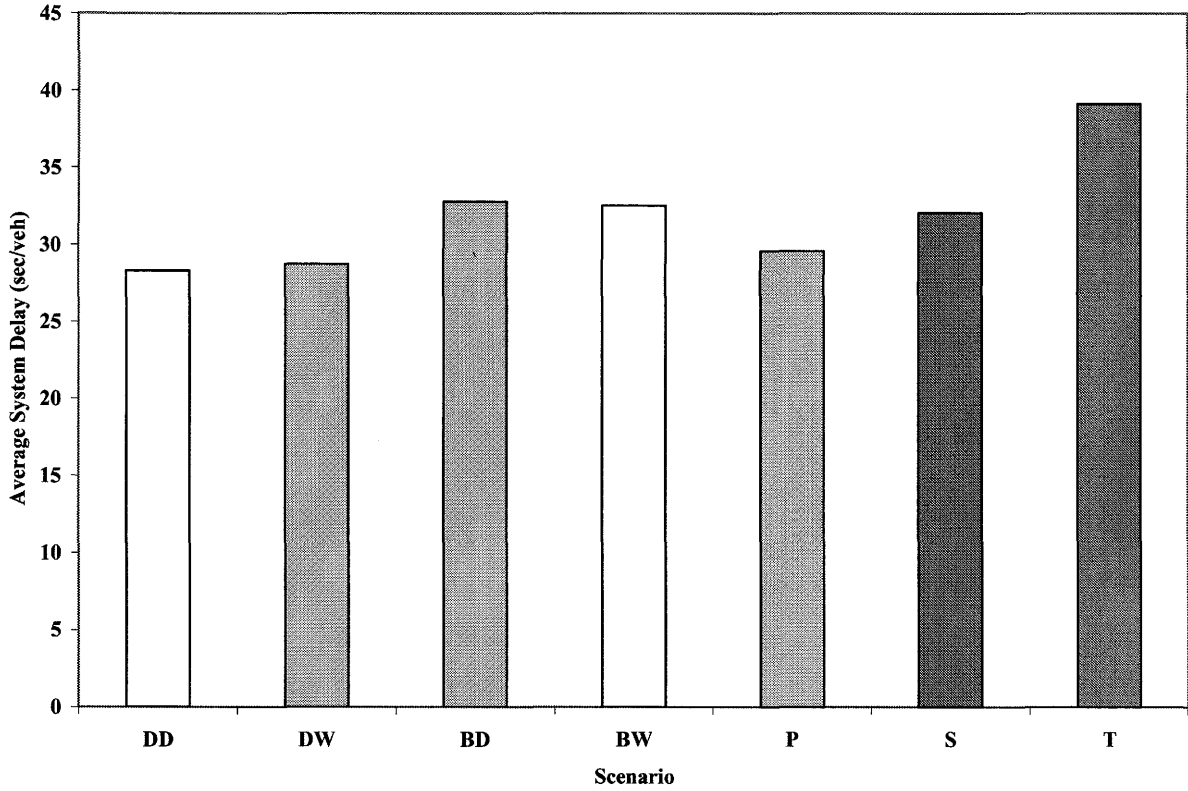
Figure 29. Delay Comparison for Best Solution for Artery 1.
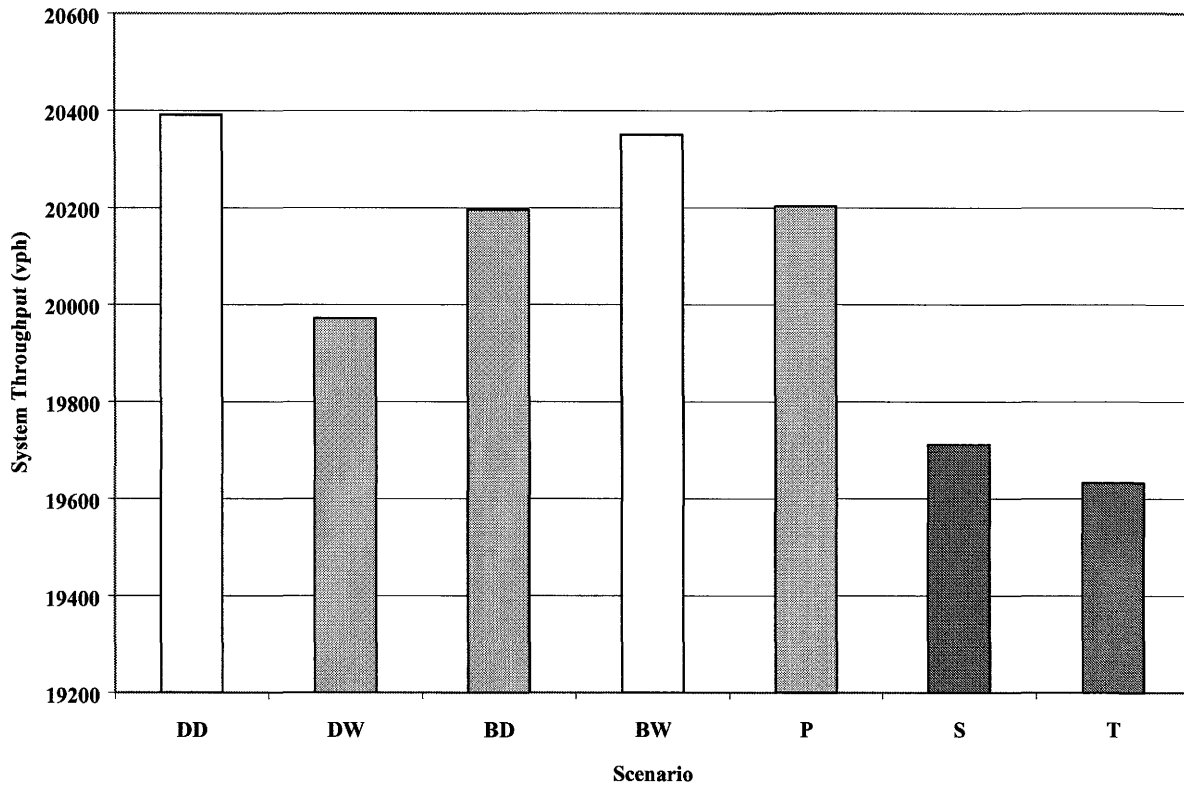
Figure 30. Throughput Comparison for Best Solution for Artery 1.
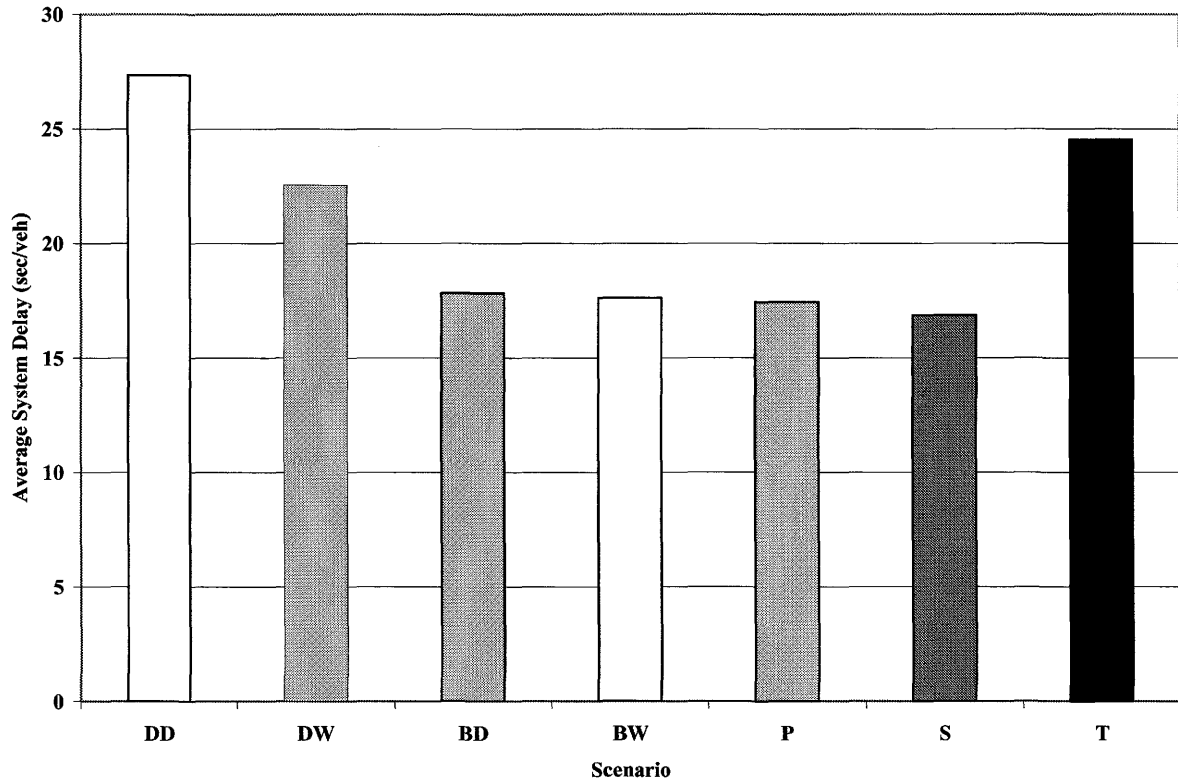
Figure 31. Delay Comparison for Best Solution for Artery 2.
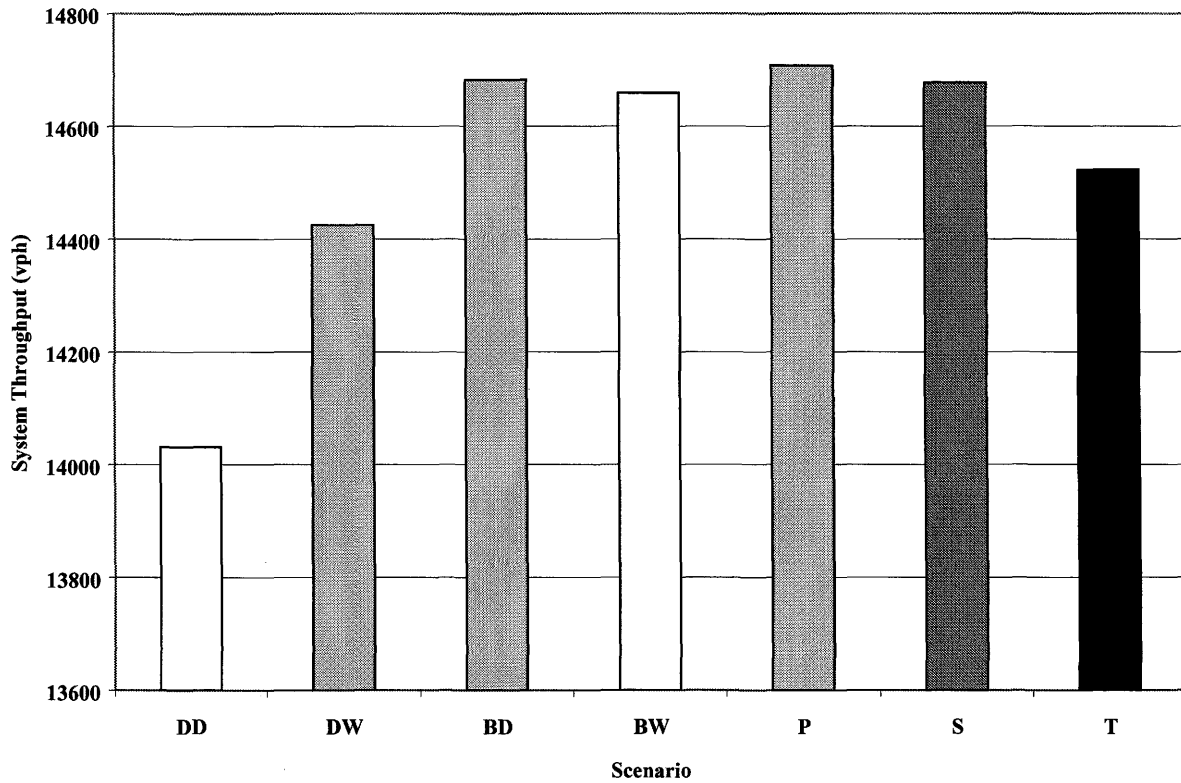
Figure 32. Throughput Comparison for Best Solution for Artery 2.
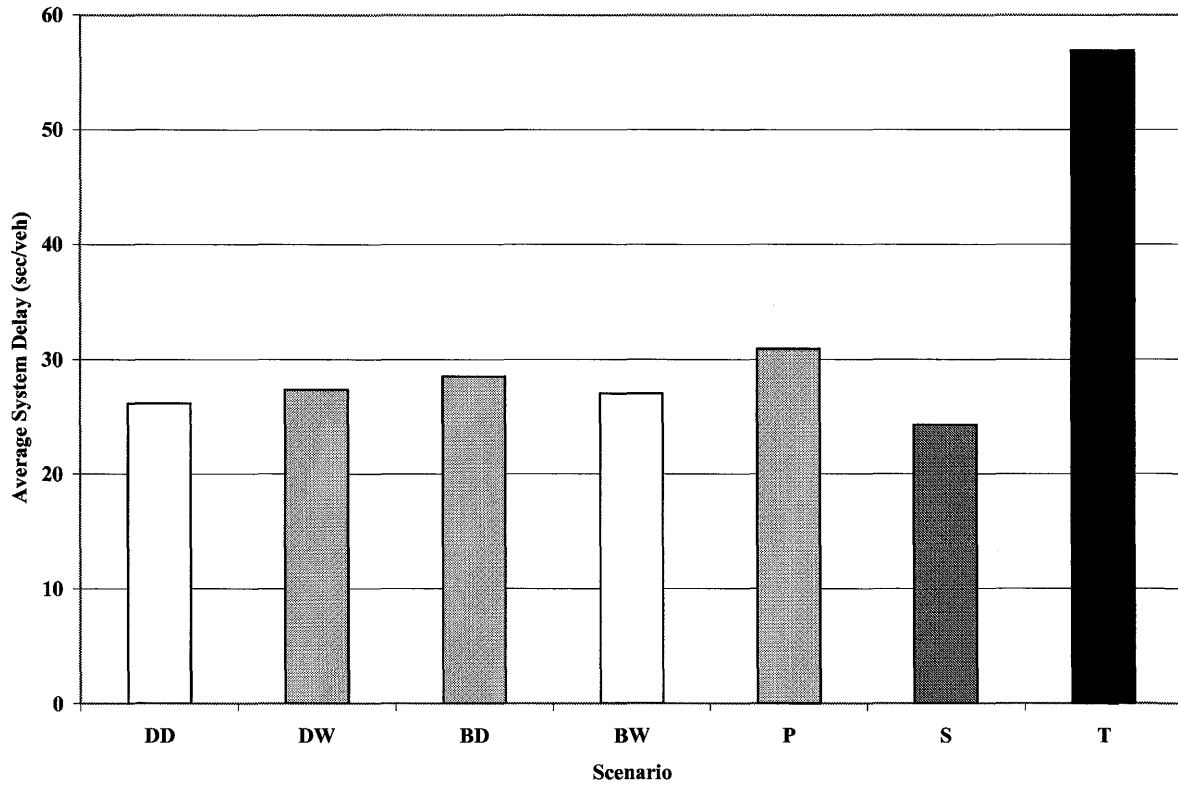
Figure 33. Delay Comparison for Best Solution for Artery 3.
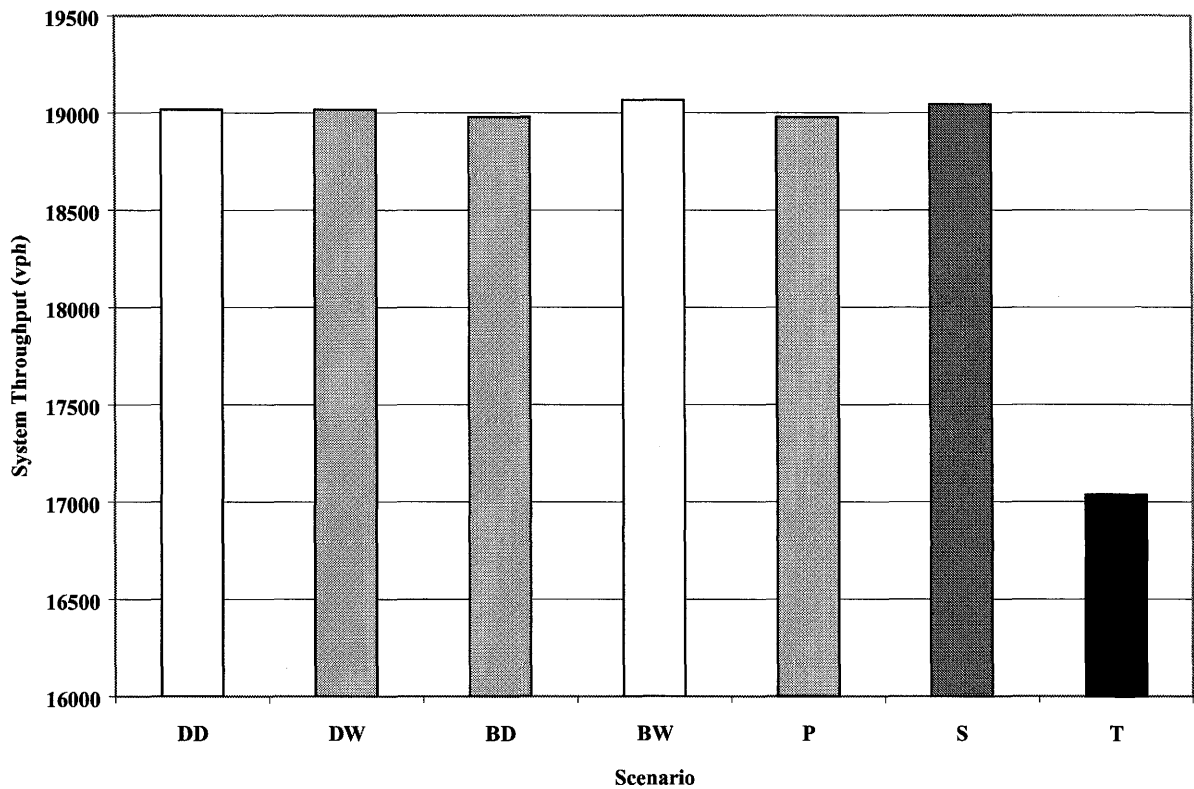
Figure 34. Throughput Comparison for Best Solution for Artery 3.

61
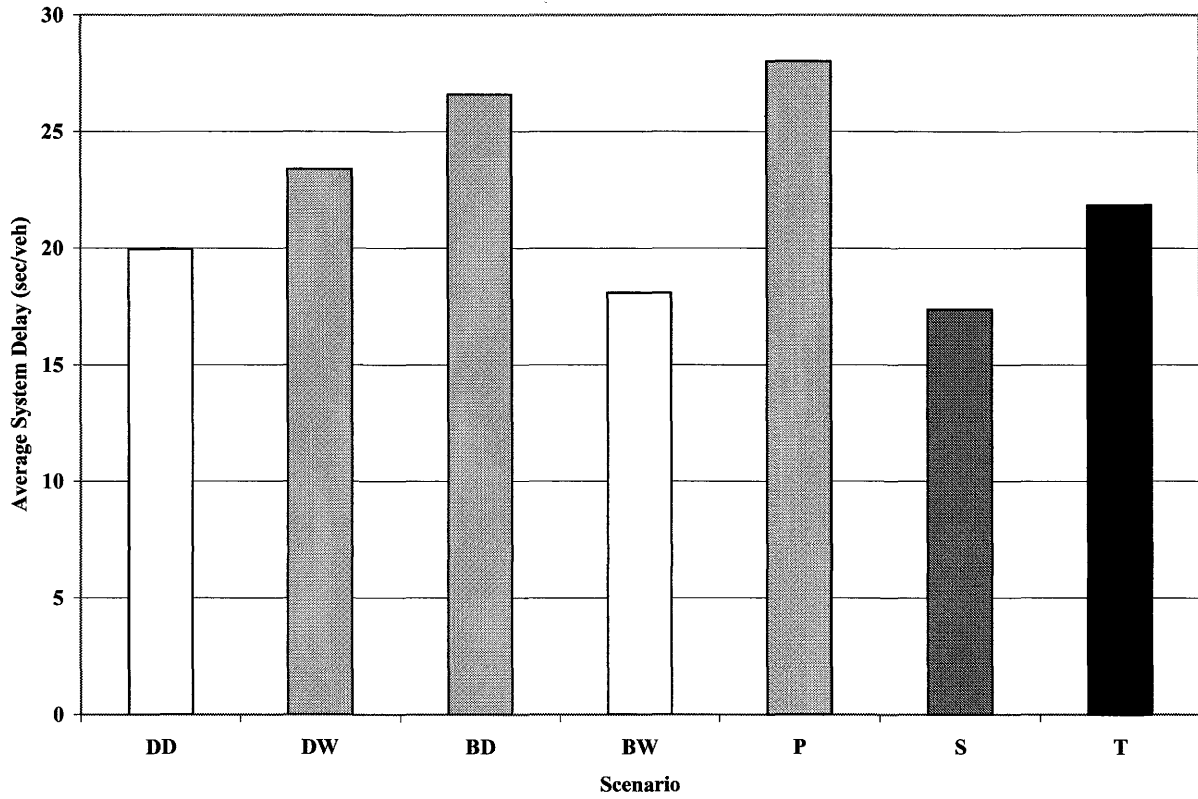
Figure 35. Delay Comparison for Best Solution for Artery 4.
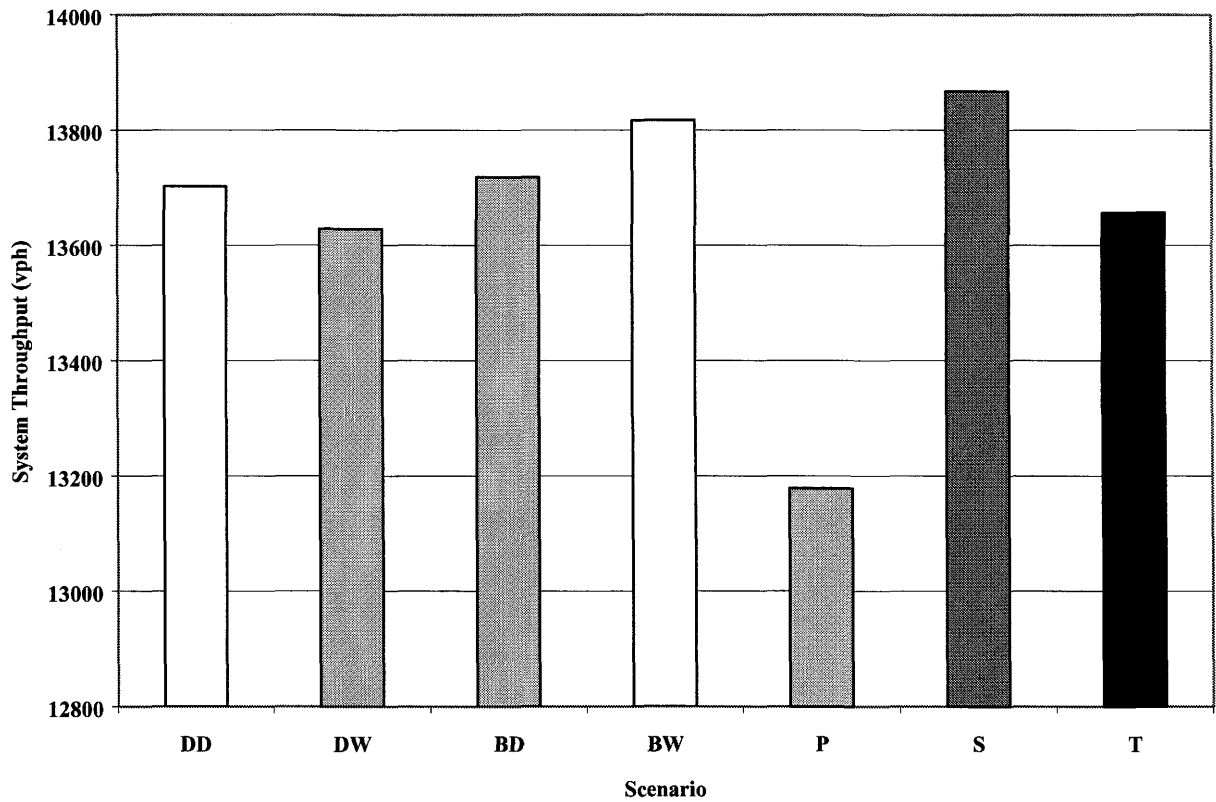
Figure 36. Throughput Comparison for Best Solution for Artery 4.

62
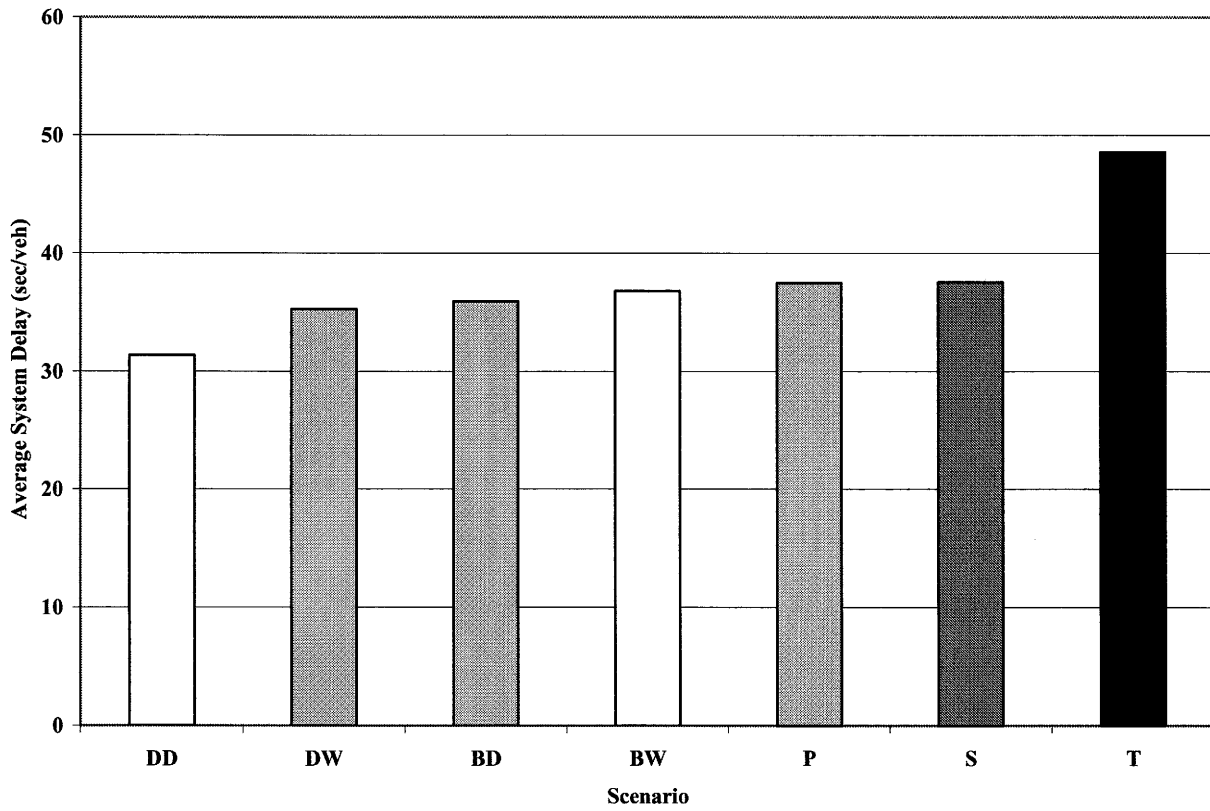
Figure 37. Delay Comparison for Best Solution for Artery 5.
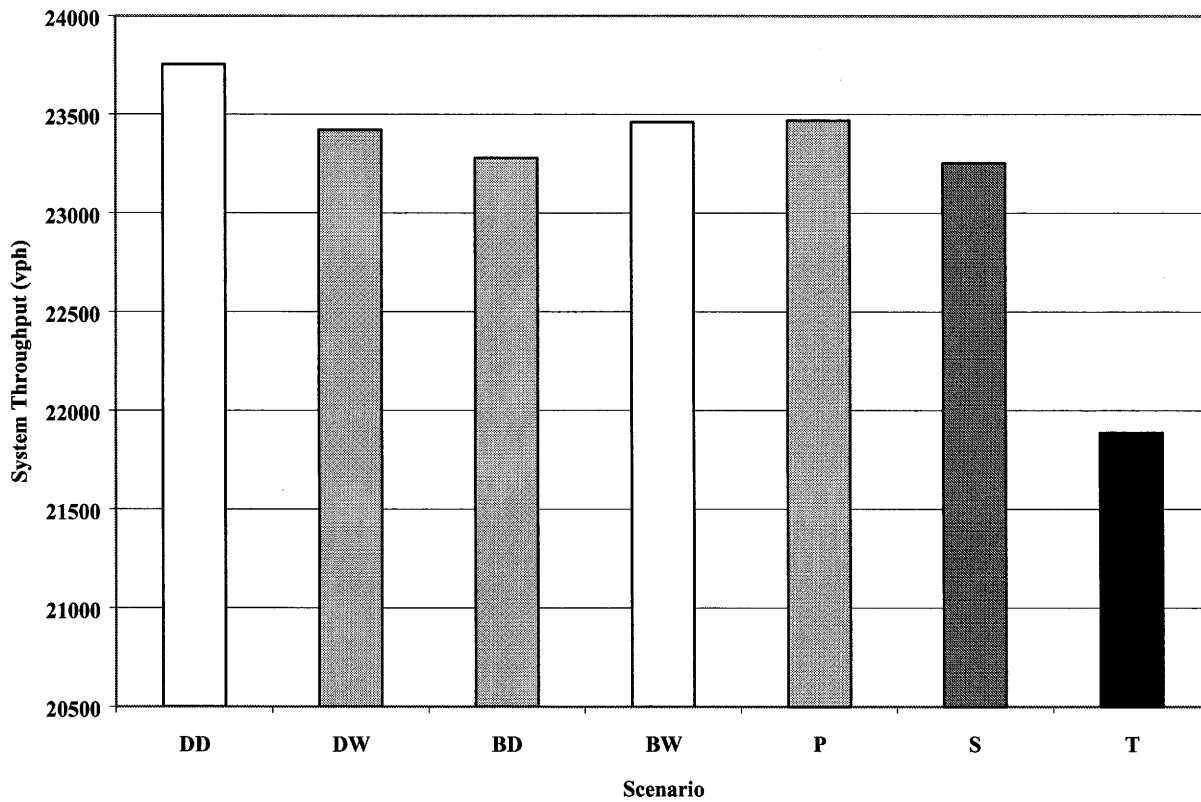
Figure 38. Throughput Comparison for Best Solution for Artery 5.

63

Table 17 provides the rank of each strategy output with respect to other strategies for both delays and throughputs. While this table has no statistical significance, it provides a feel for the relative performance of the full optimization capabilities of software compared.

Table 17. Relative Ranks of the Strategies Compared.

Strategy	Ranking for Delays						Ranking for Throughputs					
	Art 1	Art 2	Art 3	Art 4	Art 5	Mean	Art 1	Art 2	Art 3	Art 4	Art 5	Mean
DD	1	7	2	3	1	2.8	1	7	4	4	1	3.4
DW	2	5	4	5	2	3.6	5	6	3	6	4	4.8
BD	6	4	5	6	3	4.8	4	2	5	3	5	3.8
BW	5	3	3	2	4	3.4	2	4	1	2	3	2.4
P	3	2	6	7	5	4.6	3	1	6	7	2	3.8
S	4	1	1	1	6	2.6	6	3	2	1	6	3.6
T	7	6	7	4	7	6.2	7	5	7	5	7	6.2

The researchers drew the following conclusions from this analysis:

- TRANSYT 7F performed significantly worse than the other strategies.
- Synchro provided the best results for delay comparison, but the differences were not significant for all cases.
- Strategy DD was found to provide consistently good results; the only exception was Artery 2. An investigation of CORSIM results for the optimal DD timing for Artery 2 in TRAFVU revealed that lane changing blockages are formed for this timing strategy, which are not modeled in DAR.
- For some cases, we found significant differences between green-split selection methods (delay-minimized and Webster's by comparing BD to BW and DD to DW). Except for Artery 2, DD was found to provide better results than DW, while BW was found to provide better results than BD for most cases. From these results, we recommend delay minimized splits for delay analysis. We could not identify the reasons for the differences between BD and BW.
- We found that BW provided good throughput solutions for all arteries. Also, delay comparisons between the strategies show that BW provided good results.
- All progression efficiency programs (BD, BW, and PASSER II) were found to provide intermediate results. For Artery 2, all three strategies were found to provide very good results. An investigation of Artery 2 characteristics reveals a predominant arterial traffic pattern, which is conducive to progression-efficiency optimization.

Delays and throughputs form important MOEs in the evaluation of signal coordination strategies. However, the reader should note the fact that comparison of delay alone can lead to erroneous conclusions. For instance, if an approach to a link is blocked due to spillback, the link will have low-input volumes and, hence, low delays. Usually, there is a negative correlation between delays and throughputs, as can be noted from the previous figures showing software comparisons. Lastly, models that stack queues vertically may also estimate higher delays on a link by allowing more vehicles to enter the link than it has capacity to store.

SUMMARY

In this chapter, we demonstrated that GA-DAR optimization can lead to better results than PASSER III for oversaturated diamond interchanges. GA-DAR optimization also provided similar results to PASSER III in undersaturated conditions. Comparisons between GA-BAR routines and PASSER II for bandwidth optimization show that the GA-BAR routines are capable of obtaining near-optimal progression efficiency and attainability, and provide comparable results to PASSER II. Comparison of simultaneous optimization of all four signal-timing parameters with optimization of cycle length, offset, and phasing sequence with preselected green splits revealed that preselected green splits provided better results than simultaneous optimization. The results of comparisons of four PASSER V strategies with PASSER II, Synchro, and TRANSYT 7F show that Synchro and Strategy DD work best for most of the cases studied, and that all four PASSER V strategies were found to provide consistently good results. Furthermore, no single strategy was found to be best for all cases.

From earlier studies (Chapter 2), we found that bandwidth optimization of an arterial can lead to good solutions even when the arterial is experiencing oversaturated conditions if the cycle length selection considers Webster's minimum delay cycle length requirements. The results here confirm these findings and show that bandwidth optimization with the selection of the right cycle length could lead to good solutions even for oversaturated arterials. The GA-DAR routines (DD and DW) show the strength of GAs in optimization. The fitness function in GA-DAR is quite similar to TRANSYT 7F analysis, but the optimization methodology applied in TRANSYT-7F version 8.2 is a hill-climbing method. This finding is consistent with existing literature comparing GA-based optimization with the hill-climbing method.

The delay-analysis routines took between 30 minutes to 3 hours for each optimization run on a Pentium III 450 MHz with 256 MB RAM. These run times can be improved by optimizing the code, applying profilers that find bottlenecks in code execution, and applying parallel processing. These enhancements could lead to delay-based GA routines becoming viable signal- timing optimization software.

5. CONCLUSIONS AND RECOMMENDATIONS

This project compared several existing programs to analyze their performance for timing signalized arterials. As part of the project, we created PASSER V software for arterials, which contains several features and models not found in the original PASSER II program. A detailed comparison of Synchro, TRANSYT 7F, and PASSER II during the initial stages of the project used five small arterials and three volume scenarios for each. In order to obtain unbiased results, these comparisons used MOEs from CORSIM. Furthermore, researchers used average results from 20 replications of CORSIM simulations for each timing plan produced by these programs.

Then, researchers developed the new PASSER V software, which provides the best features from all of the above programs. These features include:

- a new graphic user interface,
- PASSER II's bandwidth optimization algorithm,
- a PASSER III-like model for timing diamond interchanges,
- a new delay analysis model,
- a new bandwidth analysis model,
- a GA-based optimization model for timing signals to minimize delay or maximize progression, and
- a volume/throughput analysis model.

We used real data for validating the results produced by various optimization models provided in PASSER V. Then, we compared the performance of various optimization strategies in PASSER V with PASSER II, PASSER III, Synchro, and TRANSYT 7F.

Finally, we compared newer versions of PASSER II and Synchro using three new real arterials with 10, 12, and 14 intersections, respectively. For two of these arterials, we had traffic data for two peak periods. This resulted in five scenarios.

RESEARCH FINDINGS

From the analysis of all software, we found the following about the performance of each program.

TRANSYT 7F

We used version 8.2 of this program with default objective function. It performed worse than all other programs in all aspects. Optimal timing plans produced by TRANASYT 7F provided no arterial progression bands, higher delays, and lower throughput. This version of the program does not allow phasing sequences to be optimized. Thus, the main reason for bad performance might be due to the fact that the default leading-left-turns we used for all runs may not be optimal. We also found that the cycle-length optimization algorithm in this program was inconsistent in its selection of optimal values. A delay-based program such as TRANSYT 7F should select best cycle lengths based on traffic volume levels. However, this was not the case.

Synchro

We used two versions of Synchro. In the initial stages (five small arterials), we used version 4. We found that Synchro produced plans with the lowest delay in studies where the programs were requested to optimize all signal timing parameters (cycle length, splits, offset, and phase sequences). Synchro selected best cycle lengths consistent with volume levels. In other words, it selected lower to higher cycle lengths as we increased volumes from low to high. For these arterials, it also produced good arterial progression.

At a latter stage, Synchro 5 became available. This version provides minor upgrades over the previous version. We used this new version to further analyze the program performance as compared to PASSER II for large arterials. From these studies we found that Synchro still performed best in terms of delay minimization. However, its ability to provide arterial progression degraded for these large arterials. In one case (a specific cycle length), Synchro produced larger progression bands than PASSER II. Further analysis showed that this was because of a difference in split calculations. In some cases, Synchro produced minimal or no arterial progression. Our conclusion was that Synchro does not guarantee two-way progression bands, and that any bands produced by the program are as a by-product of its objective function, which provides delay minimization with penalties for stops and queues. From these studies, we also found that Synchro has a tendency to estimate lower delays than PASSER II for the same timing plan. In addition, the delay estimates from Synchro were flat over a large range of cycle lengths.

PASSER II

In the initial stages of analysis, we compared PASSER II-90 with Synchro and TRANSYT using five small arterials. We found that PASSER II's algorithm produces the highest two-way arterial progression and system throughput and fewer stops to arterial traffic. However, the delays were higher than those for timings produced by Synchro. Higher delays resulted for PASSER II generated plans because of its tendency to select higher cycle lengths than Synchro. We also observed that the best cycle length selected by PASSER II-90 was insensitive to traffic conditions. This result is not unexpected because the bandwidth optimization methods are based on a graphical approach tied to travel times. During cycle length optimization, PASSER II-90 iterates over the cycle length range provided by the user. It first calculates splits for the smallest cycle length using Webster's method. Then it adjusts the splits to minimize delays. This method favors approaches with higher traffic demand. Thus, PASSER II allocates larger time for arterial through splits for facilities with higher arterial traffic. During cycle length optimization, the pre-calculated splits are increased proportionally to the increase in cycle length. As a result, higher and higher splits are calculated by the program for arterial through movements as cycle lengths are increased during optimization. Since the program searches for plans with maximum through bands, not efficiency, it selects higher cycle lengths.

During the next stage of analysis, we selected a range of cycle lengths based on the *Maximin* criterion. From additional analyses using these cycle lengths, we found that PASSER II still outperformed other programs in terms of bandwidth efficiency, stops, and system throughput. We also found that it outperformed Synchro 60 percent of the time by producing lower delay.

Some of PASSER II's discrepancies were resolved in PASSER II-02, released this year by TTI. In PASSER II-02, cycle length iteration has been moved from the optimization routine to the user interface. Due to this change, the optimization routine calculates unique splits for each cycle length and optimizes offsets and phasing sequences for each cycle-split set. It saves and displays results for all cycle lengths analyzed. Also, this version of the program calculates integer splits as opposed to the previous version in which splits could be fractional numbers (i.e., 10.3 seconds). Other program features have remained unchanged. We compared PASSER II-02 with Synchro 5 using three larger arterials to study the performance of these programs for such cases. From this analysis, we found that PASSER II guarantees two-way progression, even for large arterials, whereas Synchro produced no band for a significant number of cases. In one specific scenario, Synchro produced higher bands than PASSER II for the same cycle length. For this arterial and cycle length scenario, we performed further analysis by inputting Synchro's splits and phasing sequences into PASSER II. In the first run, we asked PASSER II to optimize offsets. In the second run, we asked PASSER II to optimize offsets and phasing sequences. We found that PASSER II generated larger bands than Synchro in both cases. The total band for the second scenario was larger than the first. These results verified that the difference in splits was the cause for lower PASSER II bands for this specific scenario. These studies also showed that PASSER II estimates higher delays and fuel consumption than Synchro for the same timing plan. These discrepancies are higher for lower cycle lengths. Furthermore, the differences reduce for cycle lengths above the *Maximin* cycle length. Also, the lowest delay cycle length from PASSER II was slightly higher than that from Synchro. This discrepancy is because of differences in models used by the two programs. In order to study the behavior of these models further, we conducted additional analysis using one intersection. For this case, we used several scenarios with volume-to-capacity ratios ranging from 0.5 to 1.5. These results showed that the PASSER II delay estimation model mimics the Webster's model, whereas, Synchro's model is flatter for low cycle lengths. In addition, the difference reduced for higher volume-to-capacity ratios. These results point to the fact that the models used by the two programs to estimate delay are different. From these studies, it is not evident which model is closer to reality. Field studies are needed to investigate this issue.

PASSER V

As stated earlier, we developed PASSER V in this project to improve PASSER II's ability to time arterials. PASSER V includes some of the best modeling features of all programs studied in this research. PASSER V's delay estimation model is specifically designed for all types of applications. This model is similar to TRANSYT 7F's step-wise traffic model. A delay minimization capability has also been added in the program. In addition, PASSER II's bandwidth optimization algorithm has been integrated. This implementation of the bandwidth maximization algorithm uses Webster's split without modification and removes its tendency to always select larger cycle lengths. In order to analyze the performance of algorithms provided in PASSER V, we conducted several studies using 20 replications of CORSIM simulations as in the previous cases.

First, we compared the performance of the new delay model with models in Synchro and TRANSYT 7F. We found that the new PASSER V delay model can accurately simulate queue

behavior. Its performance was found to be similar to that of TRANSYT 7F and CORSIM. However, we found that Synchro does not accurately model queue behavior. Then we compared the performance of delay-based optimization algorithms in PASSER V with PASSER III. The purpose was to determine the characteristics of this model for optimizing diamond interchanges. We found that the model's performance was similar to PASSER III for undersaturated conditions, but better and more accurate when congestion occurred on the interior link. Finally, we compared the GA-based delay and bandwidth optimization capabilities of PASSER V with PASSER II, Synchro, and TRANSYT. We found the following:

- The GA-based bandwidth optimizer in PASSER V produced comparable results to PASSER II.
- The enhanced interference minimization algorithm in PASSER V for maximizing bandwidth produces better bands and lower cycle lengths than its earlier implementations in PASSER II.
- Synchro produced slightly better delay, but the GA-based delay optimizer produces comparable results. These results could be improved by using the *Maximin* criterion. Graphical features provided in the program facilitate this selection.
- The GA-based delay optimizer performed better than other programs in terms of total throughput.

CONCLUSIONS AND RECOMMENDATIONS

PASSER V provides, under a common graphic user interface, the best features of many existing models, and it can be used by practitioners to time all types of arterials and diamond interchanges. Therefore, we recommend that this program be used by TxDOT for all future signal-timing projects. We recommend that the bandwidth-based model (preferably the *PASSER II* tool in PASSER V) be used for timing regular arterials, and the GA-based models be used for arterials with diamond interchanges. For isolated diamond interchanges, the GA-based delay minimization model should be used.

The program is equally useful for researchers in that it provides a mechanism to study the performance of several GA parameters not studied in this research.

The user interface has features to input data for multi-arterial signal networks. We recommend that the program be enhanced to include PASSER IV technology for developing progression bandwidth-based timing in multi-arterial networks.

REFERENCES

1. Webster, F.V. and B.M. Cobbe. Traffic Signals, *Road Research*, Technical Paper No. 56, Her Majesty's Stationery Office, London, England, 1966.
2. Courage, K.G. and M.R. Landmann. *Signal Operations Analysis Package*, FHWA Implementation Package 79-9, University of Florida, Gainesville, Florida, 1979.
3. CORSIM. Version 1.03, User's Manual, Kaman Science Corporation, Colorado Springs, Colorado 80933-7463, April 1997.
4. *TRANSYT 7F User's Guide*. Methodology for Optimizing Signal Timing: MOST Volume 4. University of Florida, Gainesville, Florida, March 1998.
5. Husch, D., and J. Albeck. *Synchro 4.0 User Guide*. Albany, California, 1999.
6. Chang, E.C., and C.J. Messer. *Arterial Signal Timing Optimization Using PASSER II-90, Program User's Manual*, Report 467-2F, Texas Transportation Institute, The Texas A&M University System, College Station, Texas, June 1991.
7. Messer, C.J., R.H. Whitson, C.L. Dudek, and E.J. Romano. A Variable-Sequence Multiphase Progression Optimization Program. In *Transportation Research Record 445*, TRB, National Research Council, Washington, D.C., 1973, pp. 24-33.
8. Inprise Corporation, *Borland C++ Builder 5 – Developer's Guide*, Scotts Valley, California, 2000.
9. Venglar, S., P. Koonce, and T. Urbanik II. *PASSER III-98 Application and User's Guide*, Texas Transportation Institute, The Texas A&M University System, College Station, Texas, 1998.
10. Dasgupta, D. and Z. Michalewicz. *Evolutionary Algorithms in Engineering Applications*, Springer-Verlag, New York, 1997.
11. Davis, L. *The Handbook of Genetic Algorithms*, Von Nostrand Reinhold, New York, 1991.
12. Foy, M.D., R.F. Benekohal, and D.E. Goldberg. Signal Timing Determination Using Genetic Algorithms. In *Transportation Research Record 1365*, TRB, National Research Council, Washington, D.C., 1993, pp. 108-115.
13. Wall, M. *GAlib: A C++ Library of Genetic Algorithm Components*, Version 2.4, Documentation Revision B, August 1996. http://lancet.mit.edu/ga/. Accessed October 25, 2000.
14. Chaudhary, N.A., Chi-Leung Chu, Kevin N. Balke, and Vijay G. Kovvali. *Coordination of Diamond Interchanges with Adjacent Signals*. Research Report TX-00/4913-1, Texas Transportation Institute, The Texas A&M University System, College Station, Texas, October 2000.
15. Jain, S. *An Enhanced Model for Calculating Delay as a Function of Offset*. M.S. Thesis, Texas A&M University, College Station, Texas, 1996.
16. Transportation Research Board, *Highway Capacity Manual*, TRB Special Report 209, Washington D.C., TRB, 2000.
17. Vijay G. Kovvali, *Development of a Robust Arterial Signal Coordination Software*. Ph.D. Dissertation, Department of Civil Engineering, Texas A&M University, December 2001.
18. *TSDWIN version 2.0*, http://www.fortrantraffic.com/. Accessed March 2, 2001.
19. Hadi, M.A. and C.E. Wallace. TRANSYT-7F Simulation Models for Congested Conditions, *Submitted for Presentation at the 78th Annual Meeting of the Transportation Research Board*, Washington, D.C., 1998.

20. Messer, C.J. Simulation Studies of Traffic Operations at Oversaturated, Closely Spaced Signalized Intersections. In *Transportation Research Record 1646*, Paper No. 98-0251, TRB, National Research Council, Washington, D.C., 1998, pp. 115-123.

APPENDIX A. DESCRIPTION OF DATA
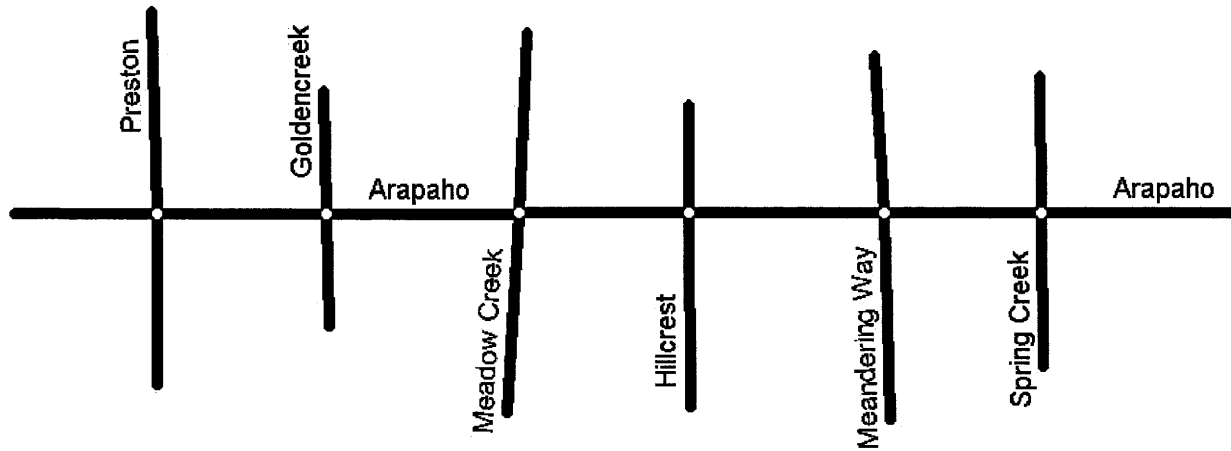
Figure A1. Sketch of Artery 1.

Table A1. Lane Assignments and Flow Data for Artery 1.

Node	Info	EBL	EBT	EBR	WBL	WBT	WBR	NBL	NBT	NBR	SBL	SBT	SBR
1	Lanes	2	3>	0	2	3>	0	1	3>	0	1	3>	0
	Flow	526	1000	167	163	697	223	189	1500	255	209	1456	219
2	Lanes	1	3>	0	1	3>	0	1	1>	0	1	1>	0
	Flow	49	1811	30	30	1059	50	30	30	30	47	30	27
3	Lanes	1	3>	0	1	3>	0	1	2>	0	1	2>	0
	Flow	119	1729	36	98	1049	115	30	80	91	73	46	60
4	Lanes	1	3>	0	1	3>	0	1	3>	0	1	3>	0
	Flow	176	1500	170	400	1312	113	200	1000	107	233	642	141
5	Lanes	1	3>	0	1	3>	0	1	2>	0	1	2>	0
	Flow	95	1456	48	37	1300	55	50	253	68	35	118	40
6	Lanes	1	3>	0	1	3>	0	1	1>	0	1	1>	0
	Flow	34	1479	42	44	966	32	32	40	64	27	30	30

EBL: Eastbound Left	WBL: Westbound Left	NBL: Northbound Left	SBL: Southbound Left
EBT: Eastbound Through	WBT: Westbound Through	NBT: Northbound Through	SBT: Southbound Through
EBR: Eastbound Right	WBR: Westbound Right	NBR: Northbound Right	SBR: Southbound Right
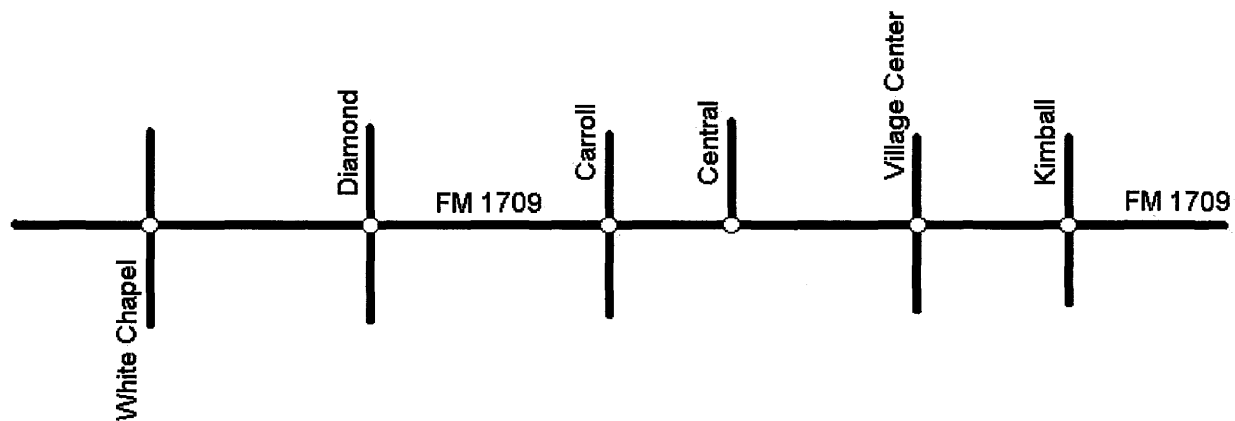
Figure A2. Sketch of Artery 2.

Table A2. Lane Assignments and Flow Data for Artery 2.

Node	Info	EBL	EBT	EBR	WBL	WBT	WBR	NBL	NBT	NBR	SBL	SBT	SBR
1	Lanes	1	2>	0	1	3	1	0	1>	0	1	1	0
	Flow	88	1000	0	86	2000	165	0	100	49	122	125	0
2	Lanes	1	2	1	1	3	1	1	1>	0	1	1>	0
	Flow	30	1138	104	149	1934	50	92	30	149	30	30	0
3	Lanes	1	2	1	1	2	1	0	<1>	0	1	1	1
	Flow	91	1212	50	101	1400	149	53	48	57	57	59	110
4	Lanes	1	2			2	1				1		1
	Flow	30	1269			2173	30				48		30
5	Lanes	1	2	1	1	2	1	1	1>	0	1	1	1
	Flow	112	1221	0	30	1781	95	30	30	30	200	0	237
6	Lanes	1	2	1	1	2	1	1	1>	0	1	1	1
	Flow	135	1105	39	53	1570	74	56	47	28	79	50	264
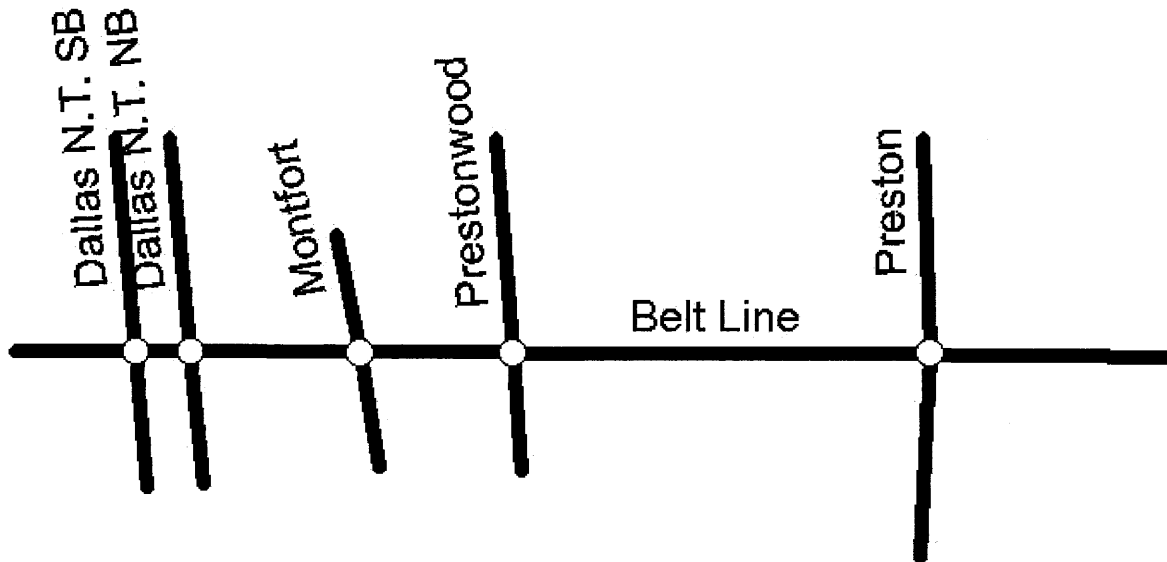
Figure A3. Sketch of Artery 3.

Table A3. Lane Assignments and Flow Data for Artery 3.

Node	Info	EBL	EBT	EBR	WBL	WBT	WBR	NBL	NBT	NBR	SBL	SBT	SBR
1	Lanes	0	4>	0	1	3	0	0	0	0	1	3	1
1	Flow	0	2126	438	322	1641	0	0	0	0	245	784	257
2	Lanes	1	3	0	0	4	1	0	<3	1	0	0	0
2	Flow	400	1762	0	0	1535	276	454	1000	400	0	0	0
3	Lanes	2	3>	0	2	3>	0	1	2	1	1	2>	0
3	Flow	224	1880	80	218	1261	86	240	204	148	105	227	118
4	Lanes	1	3>	0	1	4>	0	1	1>	0	2	1>	1
4	Flow	348	2172	10	30	1300	171	30	30	30	204	30	180
5	Lanes	2	3	1	1	3>	0	2	3>	0	1	3	1
5	Flow	300	1432	178	65	500	129	593	1800	78	159	1277	102
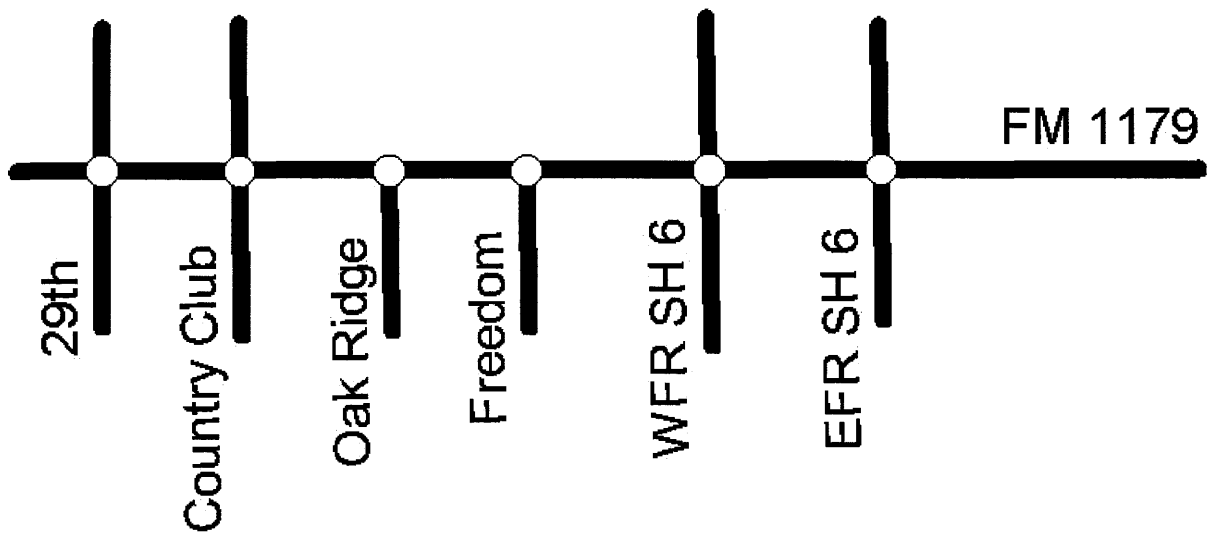
Figure A4. Sketch of Artery 4.

Table A4. Lane Assignments and Flow Data for Artery 4.

Node	Info	EBL	EBT	EBR	WBL	WBT	WBR	NBL	NBT	NBR	SBL	SBT	SBR
1	Lanes	1	2	1	1	2	1	2	2	1	2	2	1
1	Flow	74	1043	141	193	714	408	162	536	351	498	509	49
2	Lanes	1	2	1	1	2>	0	1	1>	0	1	1>	0
2	Flow	48	1679	31	30	1251	17	44	30	29	50	30	50
3	Lanes		2>	0	1	2		1		1			
3	Flow		1716	60	102	1261		131		49			
4	Lanes		2>	0	1	2		2		1			
4	Flow		1447	160	178	1111		252		335			
5	Lanes	0	2>	0	1	2	0	0	0	0	1	2	1
5	Flow	0	900	400	150	1122	0	0	0	0	88	171	200
6	Lanes	1	2	0	0	2	1	2	1	1	0	0	0
6	Flow	663	748	0	0	529	63	693	65	250	0	0	0
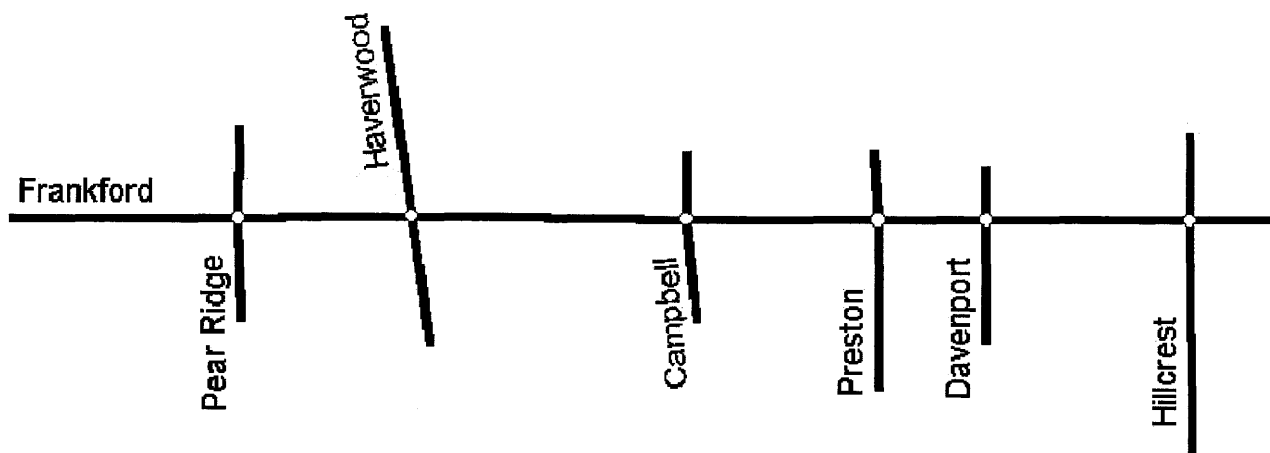
Figure A5. Sketch of Artery 5.

Table A5. Lane Assignments and Flow Data for Artery 5.

Node	Info	EBL	EBT	EBR	WBL	WBT	WBR	NBL	NBT	NBR	SBL	SBT	SBR
1	Lanes	1	3	0	0	3>	0	0	0	0	1	0	1
	Flow	241	1802	0	0	1829	442	0	0	0	260	0	102
2	Lanes	1	3>	0	1	3>	0	1	1>	0	1	1>	0
	Flow	55	1907	51	76	2225	30	37	30	33	30	30	30
3	Lanes	1	3>	0	1	3>	0	1	2>	0	1	1>	0
	Flow	218	1442	202	89	1626	40	200	91	75	37	73	107
4	Lanes	2	3>	0	2	3>	0	1	3>	0	1	3>	0
	Flow	428	1029	114	152	875	147	448	2283	341	208	1222	368
5	Lanes	1	3>	0	1	3>	0	1	2>	0	1	1>	0
	Flow	58	1402	93	74	1076	42	91	63	104	30	43	30
6	Lanes	1	3>	0	1	3>	0	1	2	1	1	3>	0
	Flow	159	1184	163	72	1005	197	221	748	88	128	297	90

Artery	Link 1	Link 2	Link 3	Link 4	Link 5
1	1834	2073	1847	2105	1702
2	2500	2700	1390	2110	1700
3	386	1183	1067	2915	n/a
4	820	890	820	1085	1025
5	2229	3528	2448	1389	2620
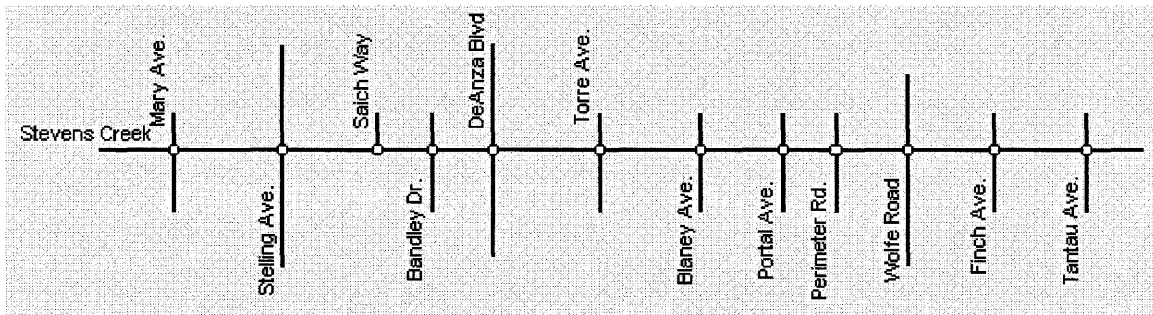
Figure A6. Sketch of Artery 6.
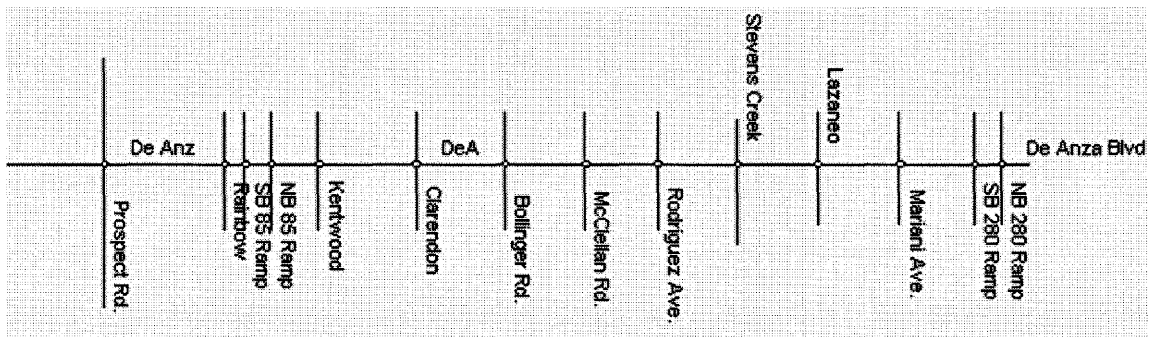
Figure A7. Sketch of Artery 7.
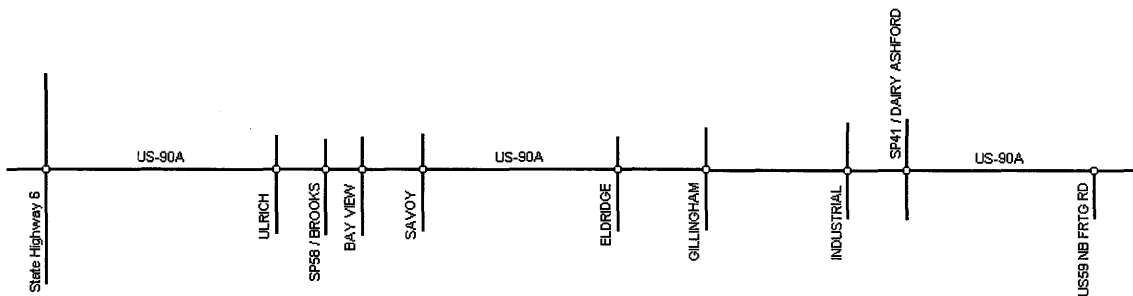
Figure A8. Sketch of Artery 8.

APPENDIX B. ANALYSIS RESULTS
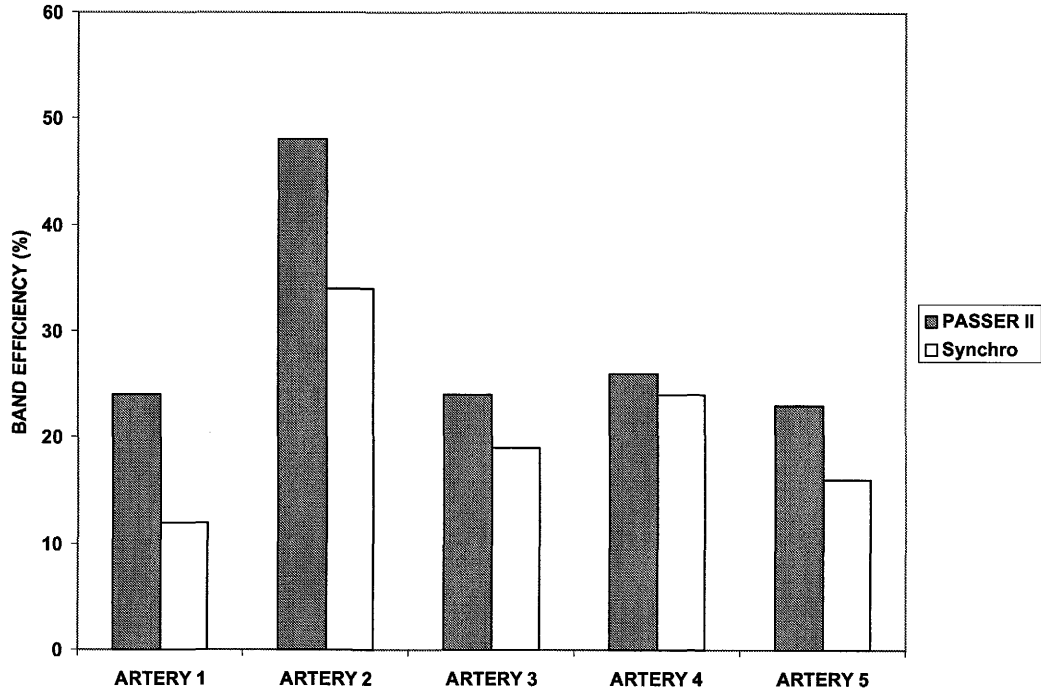
Figure B1. Best Solution Bandwidth Efficiency for 70 Percent Volume Scenario.
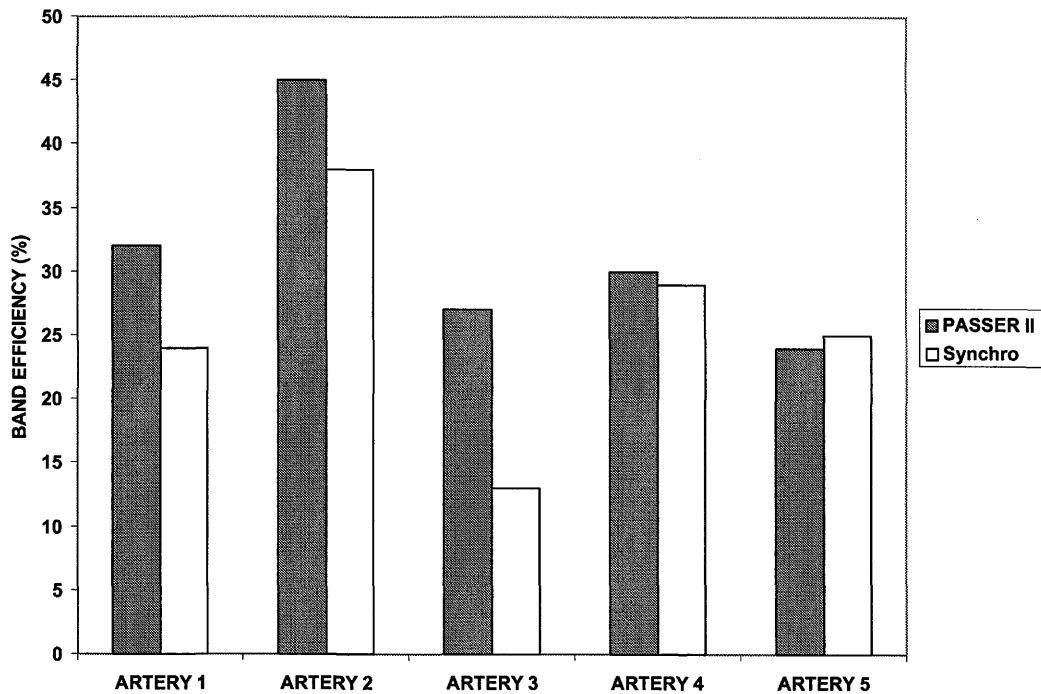
Figure B2. Best Solution Bandwidth Efficiency for the Base Volume Scenario.
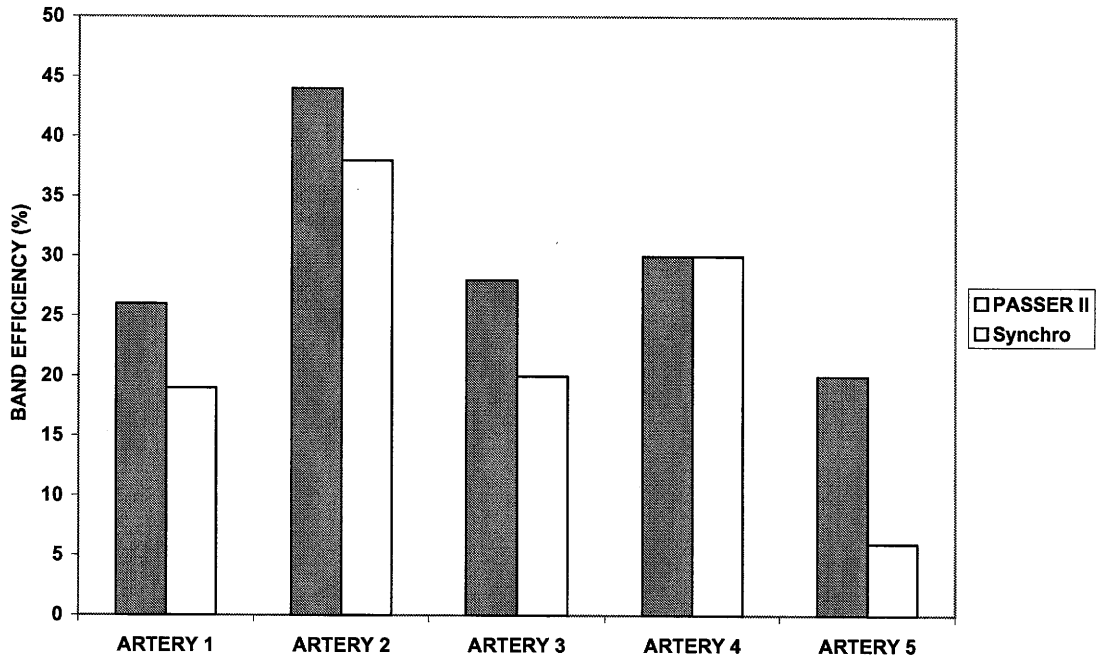
Figure B3. Best Solution Band Efficiency for the 120 Percent Volume Scenario.
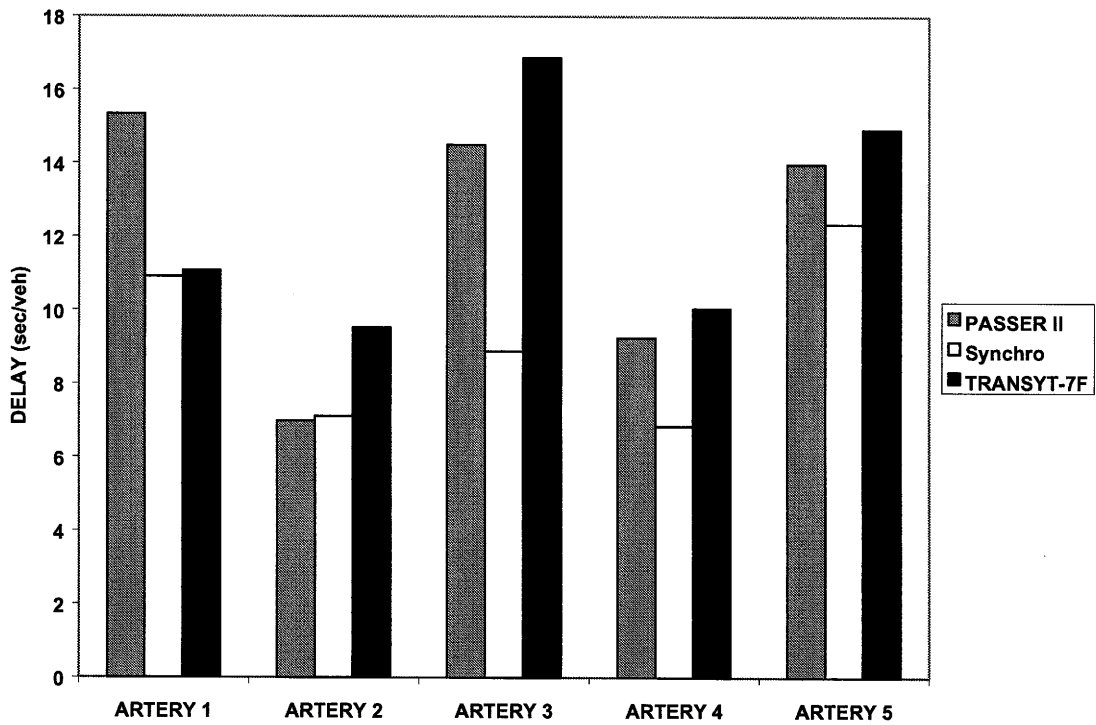
Figure B4. Comparison of Delay per Vehicle for the Best Solutions (GF=0.7).

80
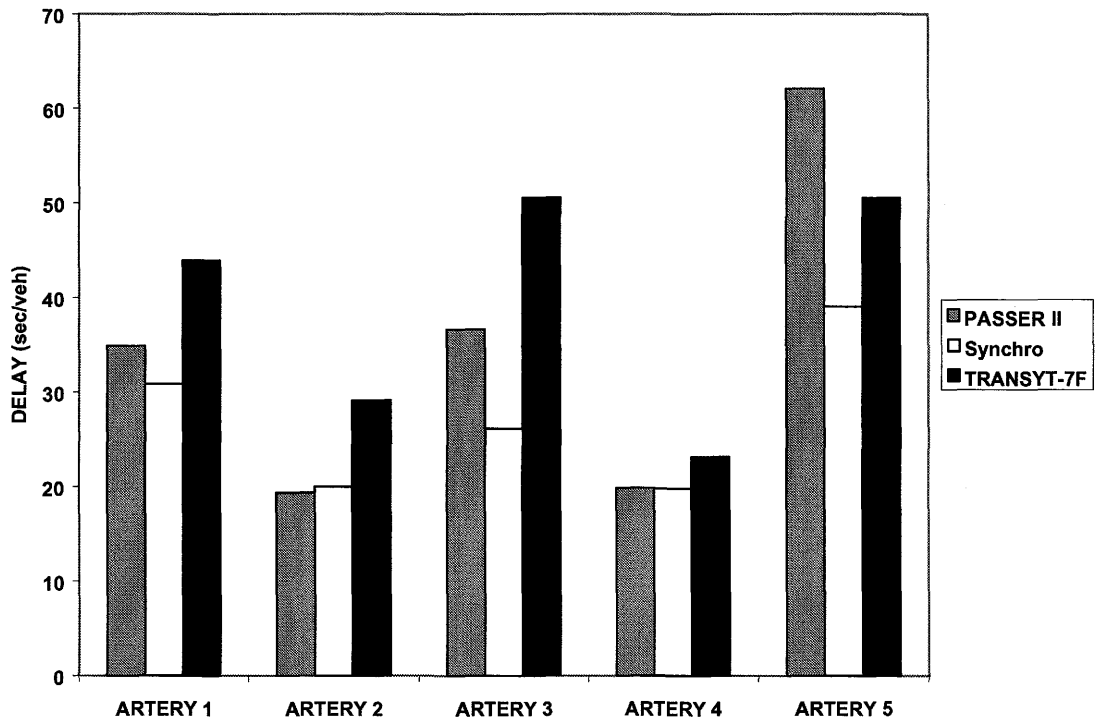
Figure B5. Comparison of Delay per Vehicle for the Best Solutions (GF=1.0).
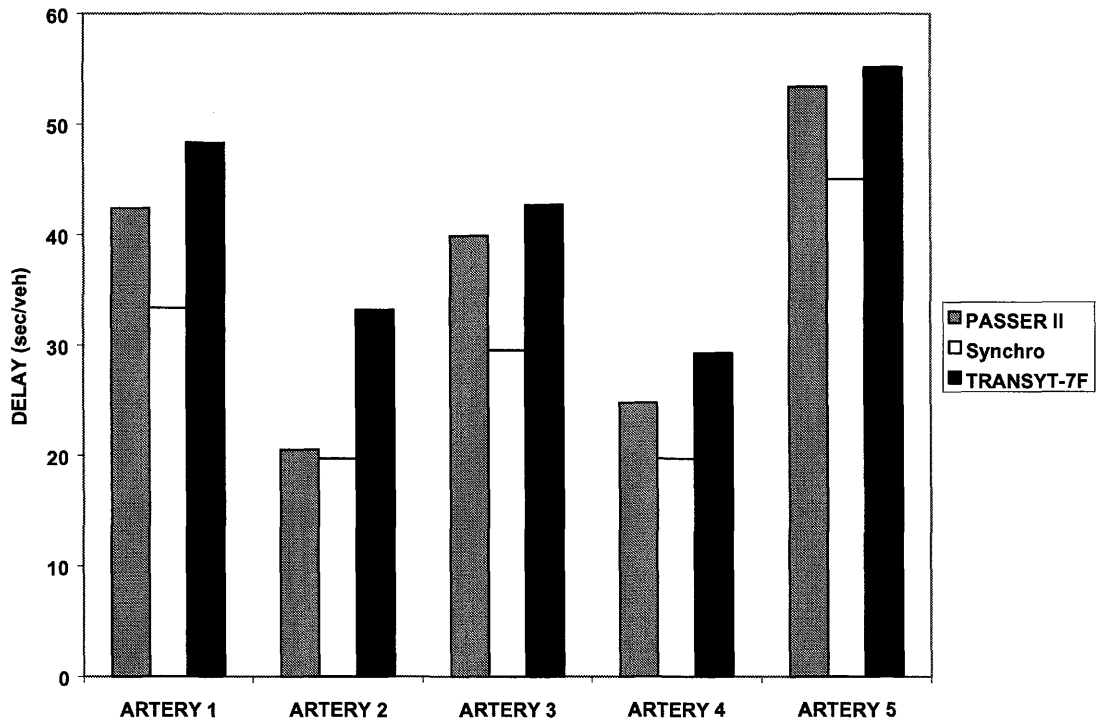
Figure B6. Comparison of Delay per Vehicle for the Best Solutions (GF=1.2).
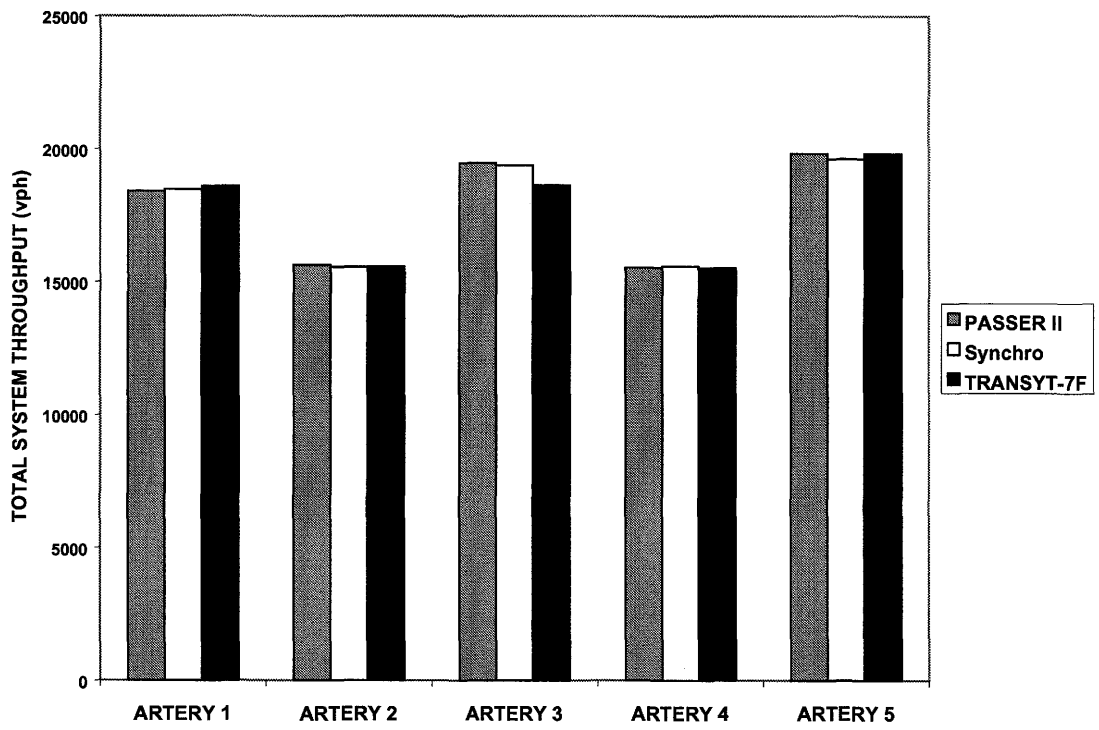
Figure B7. Comparison of System Throughput for the Best Solutions (GF=0.7).
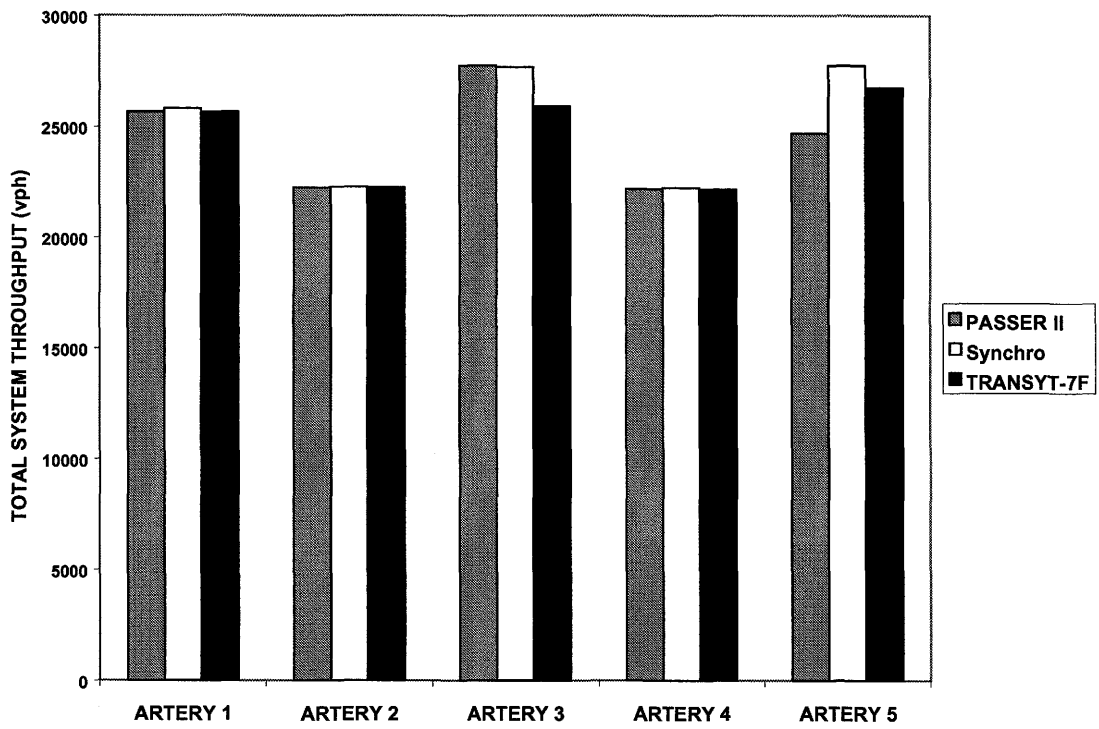
Figure B8. Comparison of System Throughput for the Best Solutions (GF=1.0).
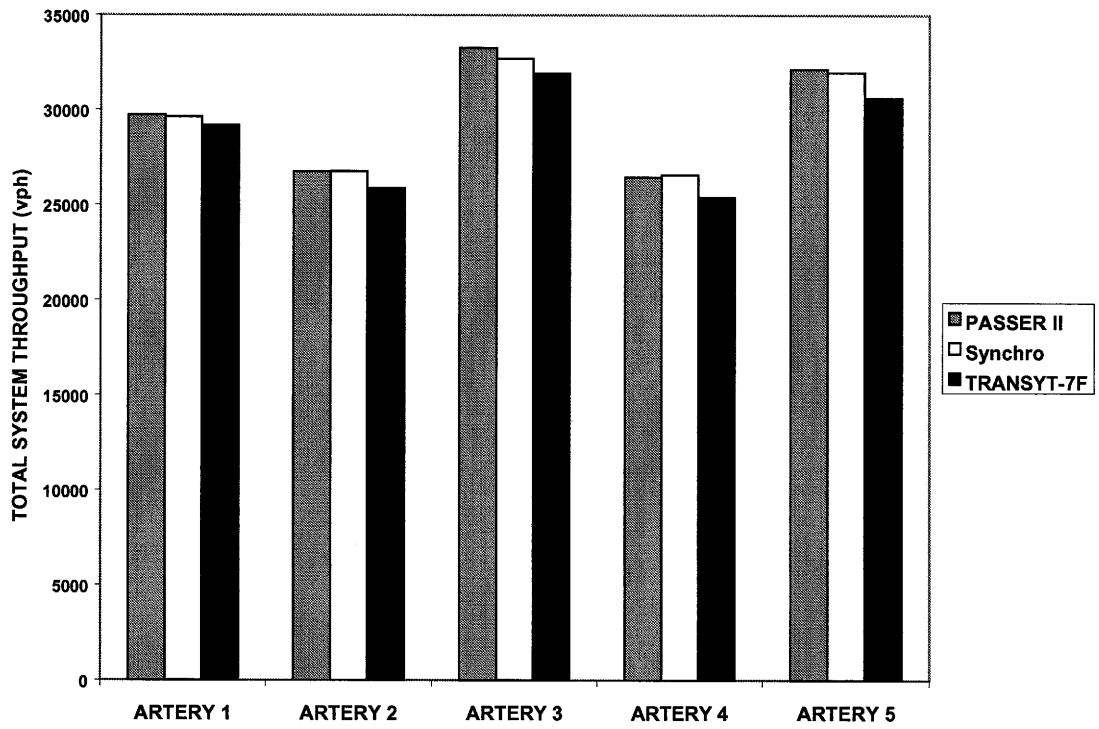
Figure B9. Comparison of System Throughput for the Best Solutions (GF=1.2).
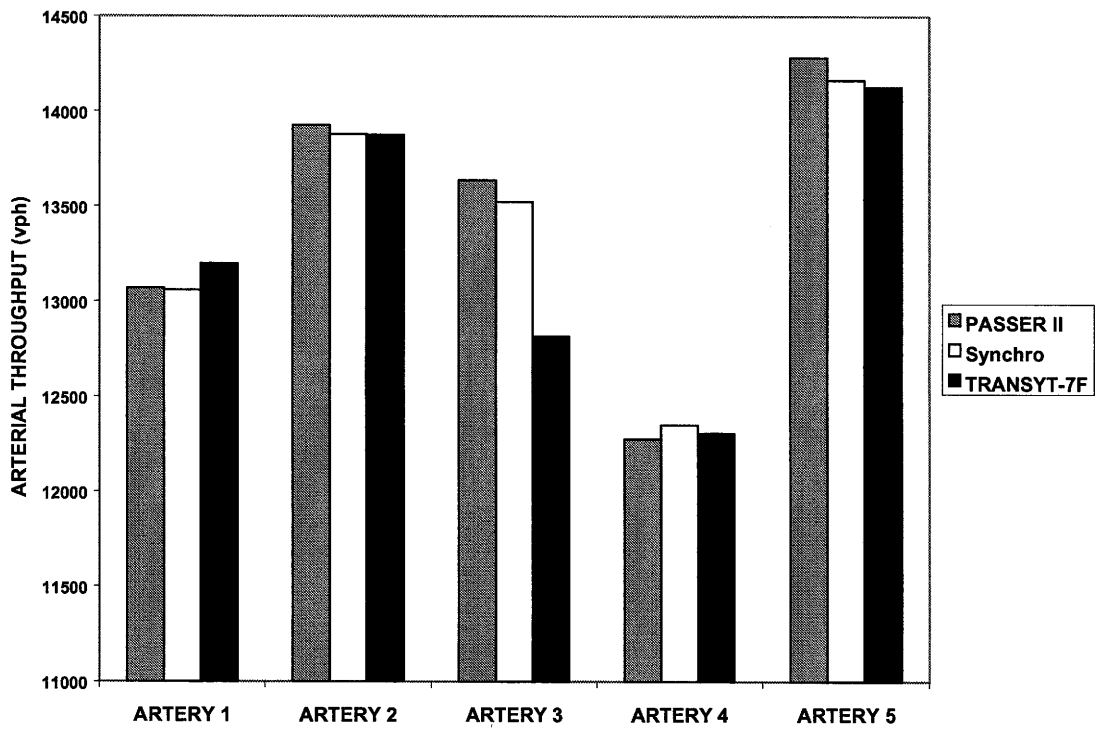
Figure B10. Comparison of Arterial Throughput for the Best Solutions (GF=0.7).
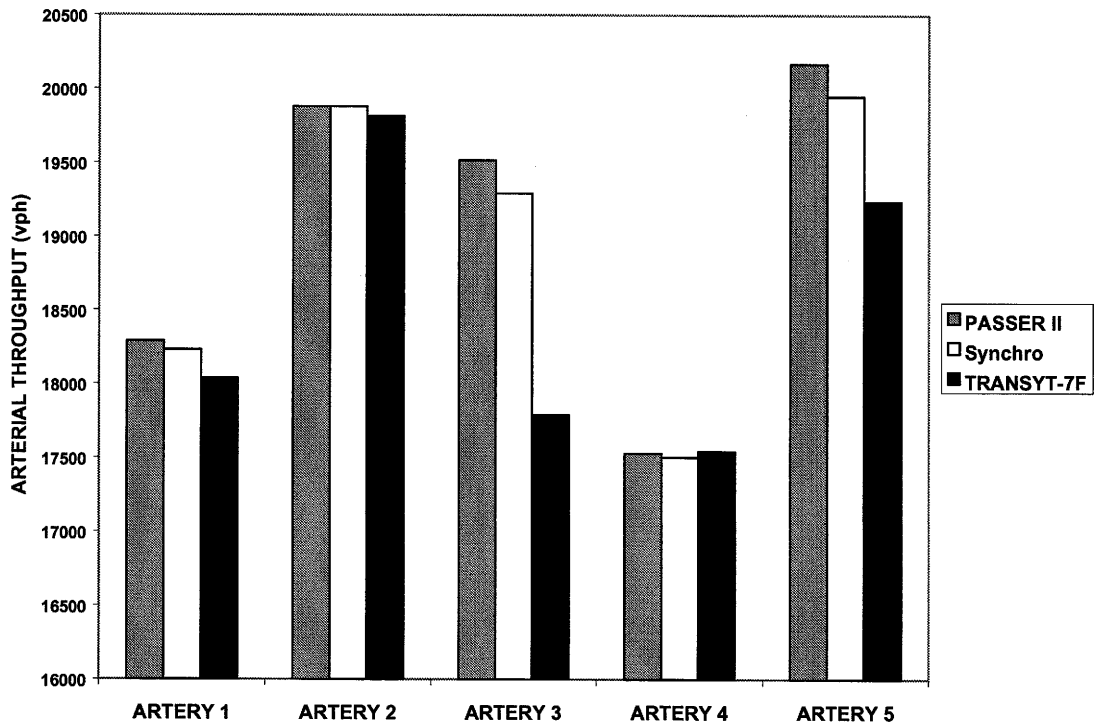
Figure B11. Comparison of Arterial Throughput for the Best Solutions (GF=1.0).
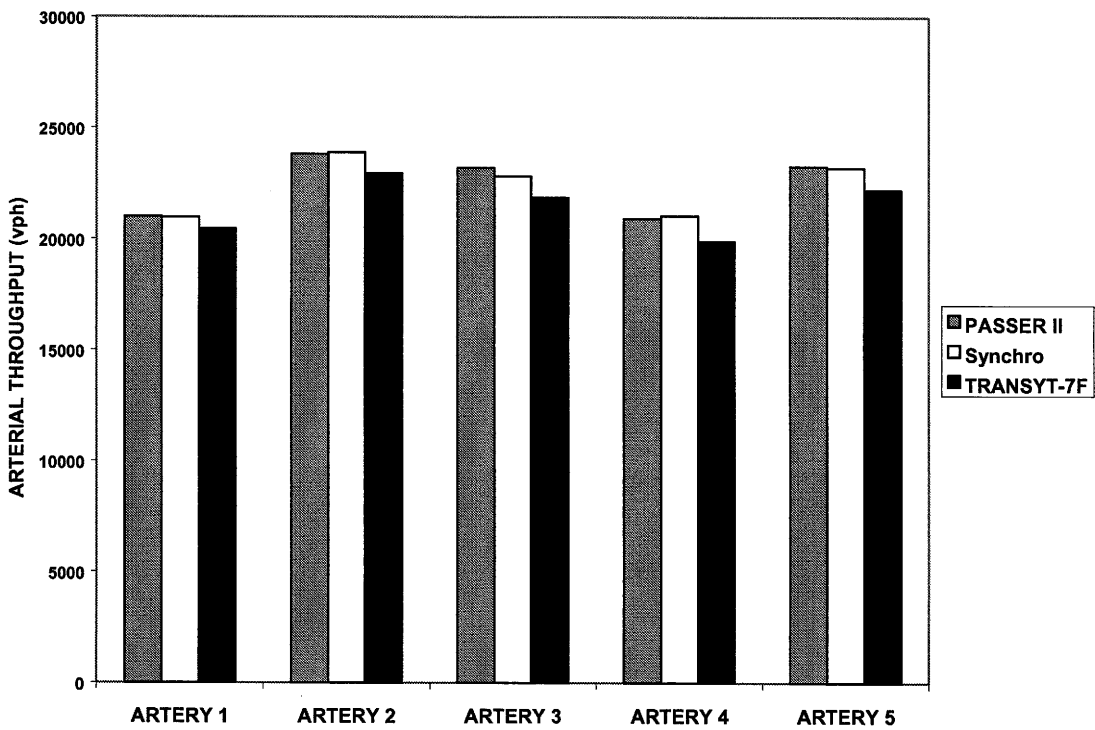
Figure B12. Comparison of Arterial Throughput for the Best Solutions (GF=1.2).
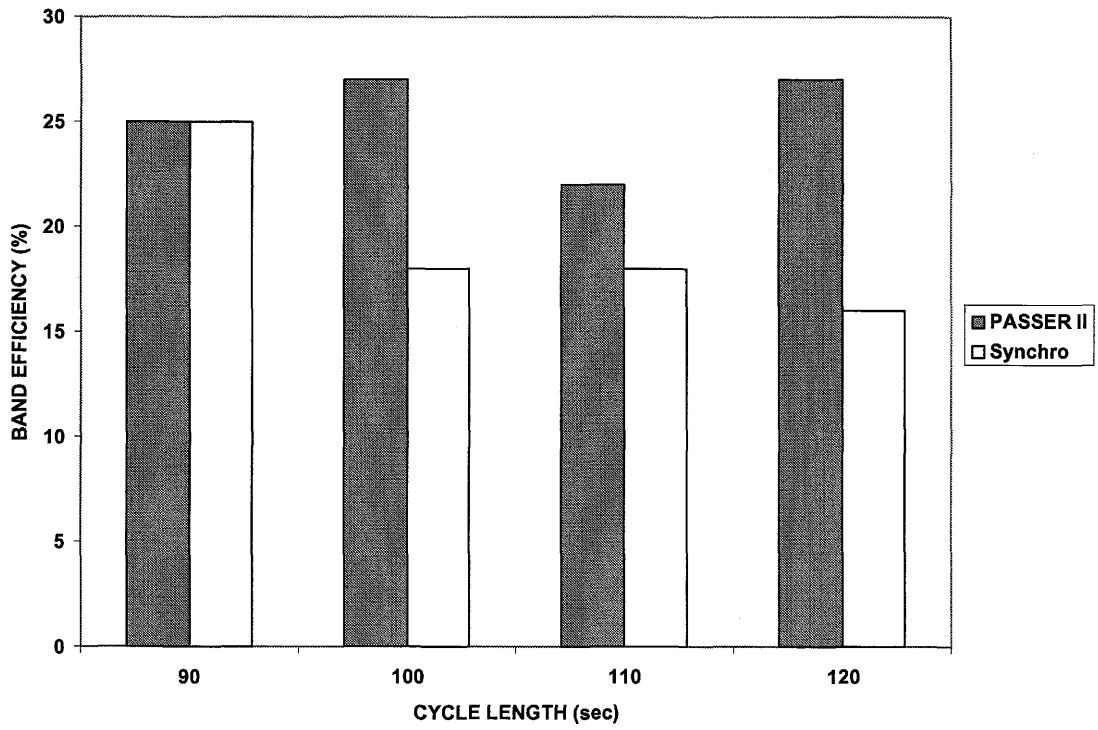
Figure B13. Synchro versus PASSER II Bandwidth Efficiency for Artery 1.
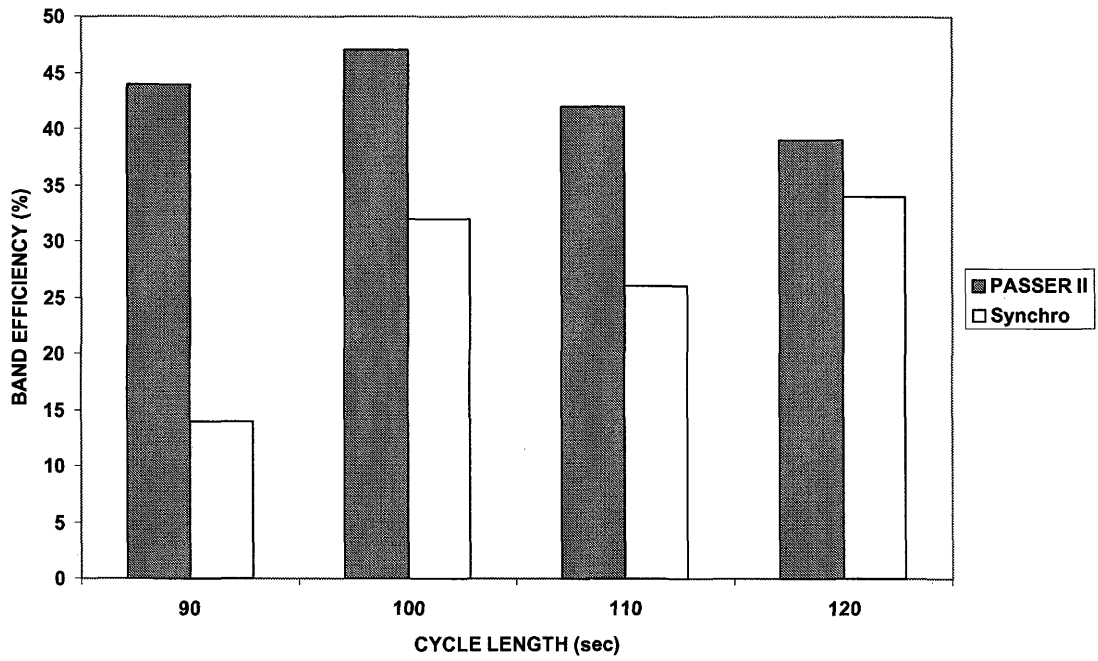
Figure B14. Synchro versus PASSER II Bandwidth Efficiency for Artery 2.
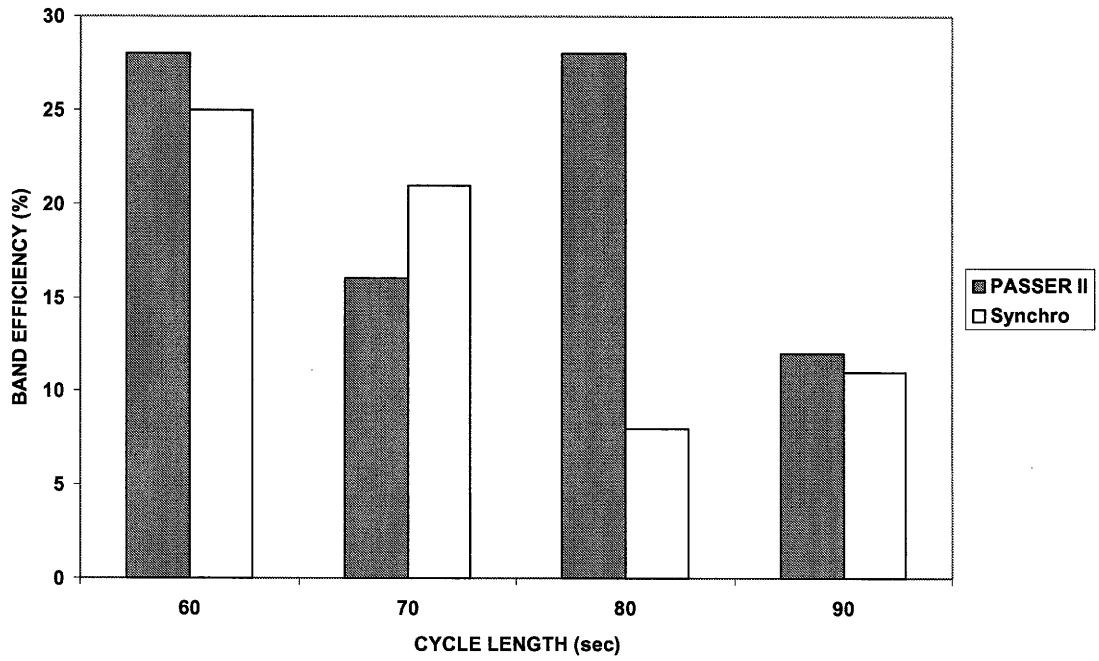
Figure B15. Synchro versus PASSER II Bandwidth Efficiency for Artery 3.
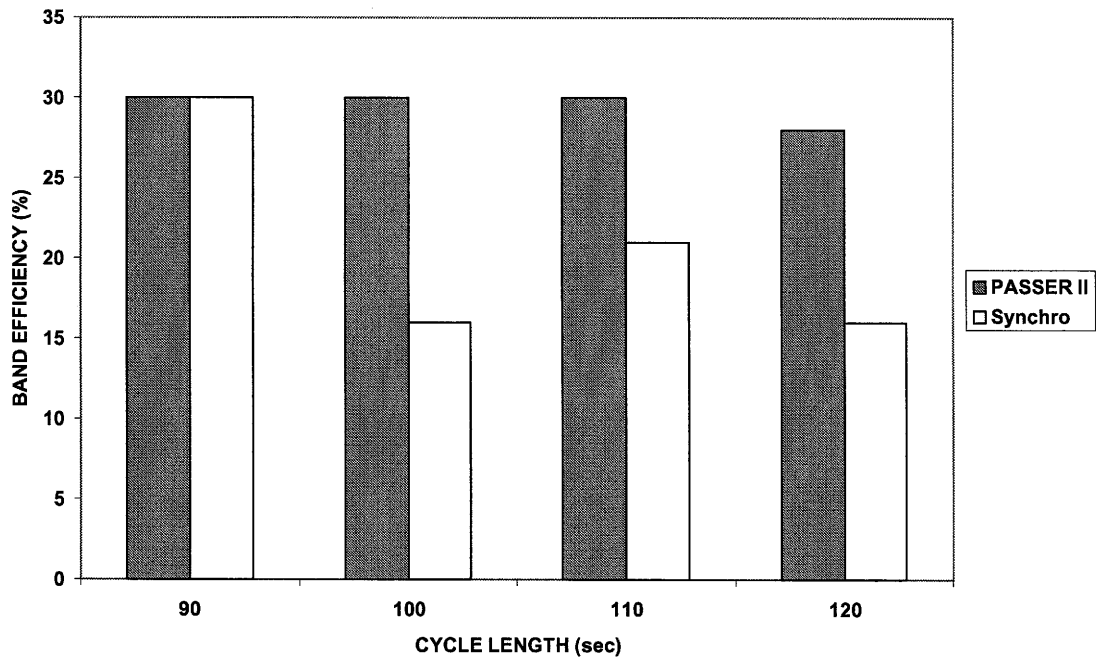
Figure B16. Synchro versus PASSER II Bandwidth Efficiency for Artery 4.

86
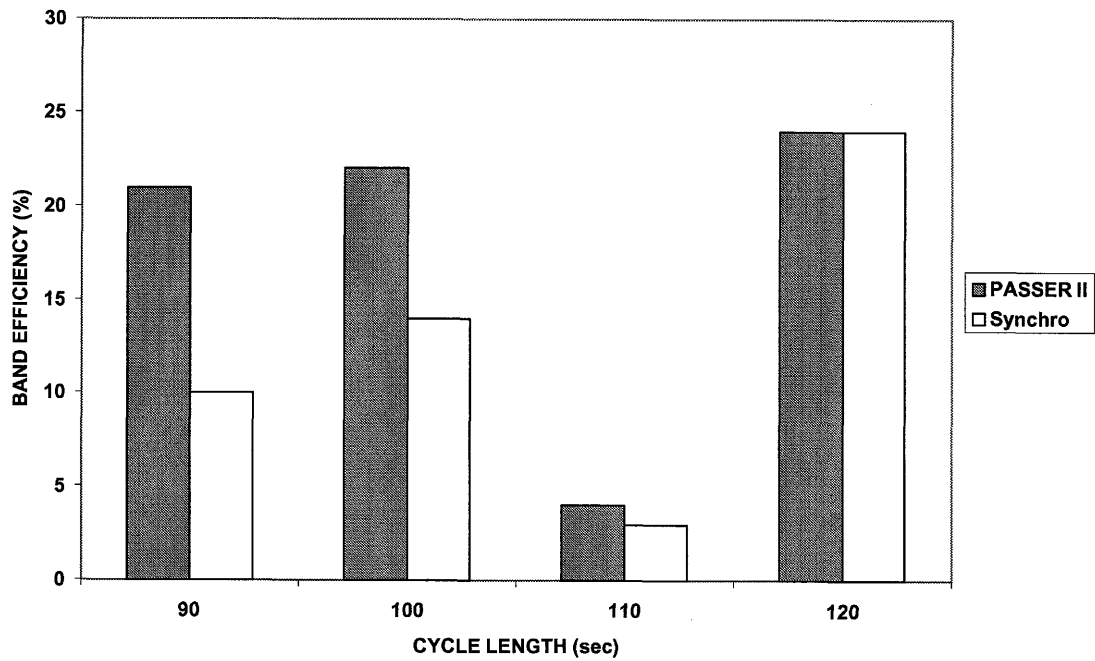
Figure B17. Synchro versus PASSER II Bandwidth Efficiency for Artery 5.
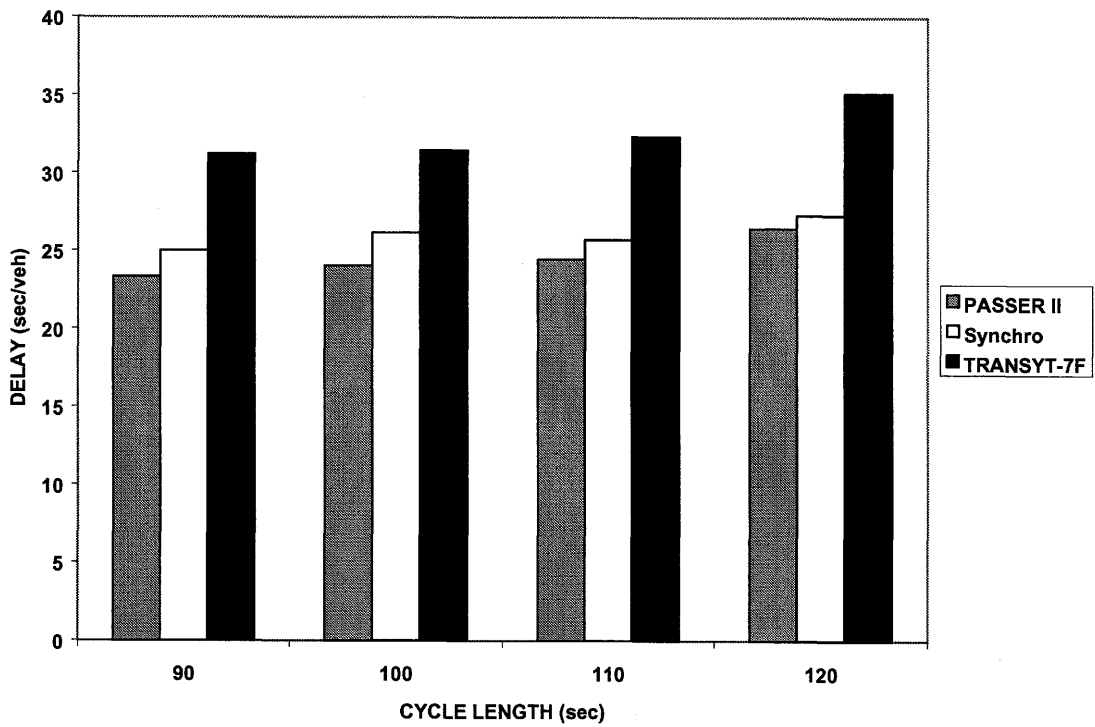
Figure B18. Comparison of Delay per Vehicle by Cycle Length for Artery 1 (GF=1.0).

87
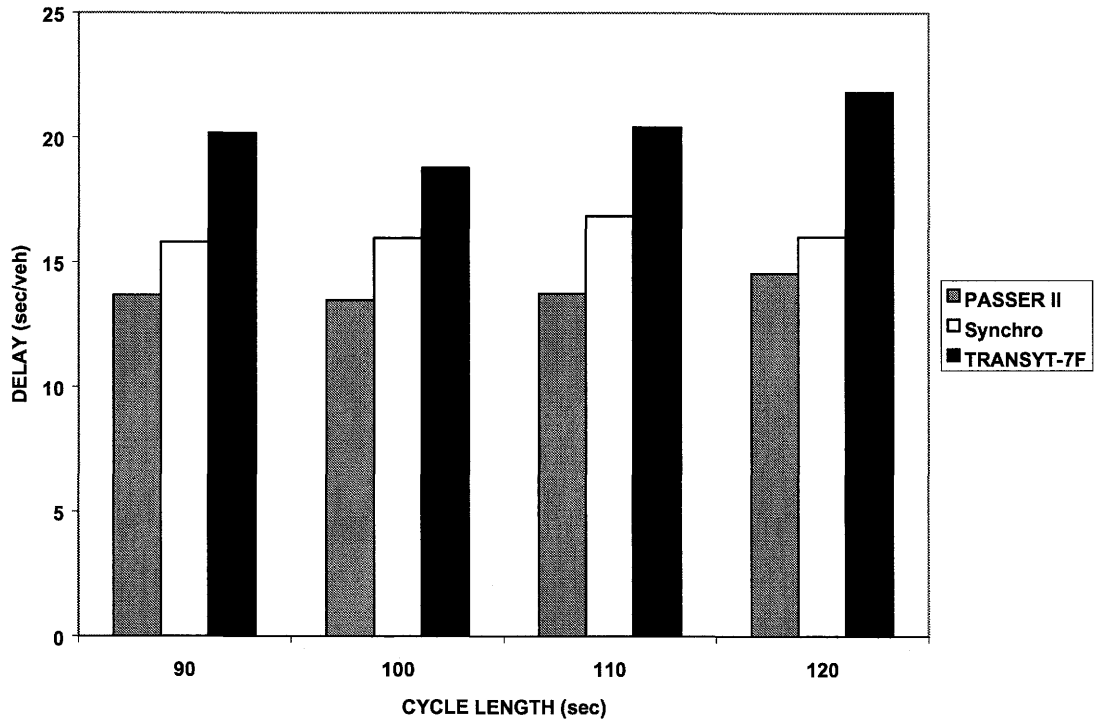
Figure B19. Comparison of Delay per Vehicle by Cycle Length for Artery 2 (GF=1.0).
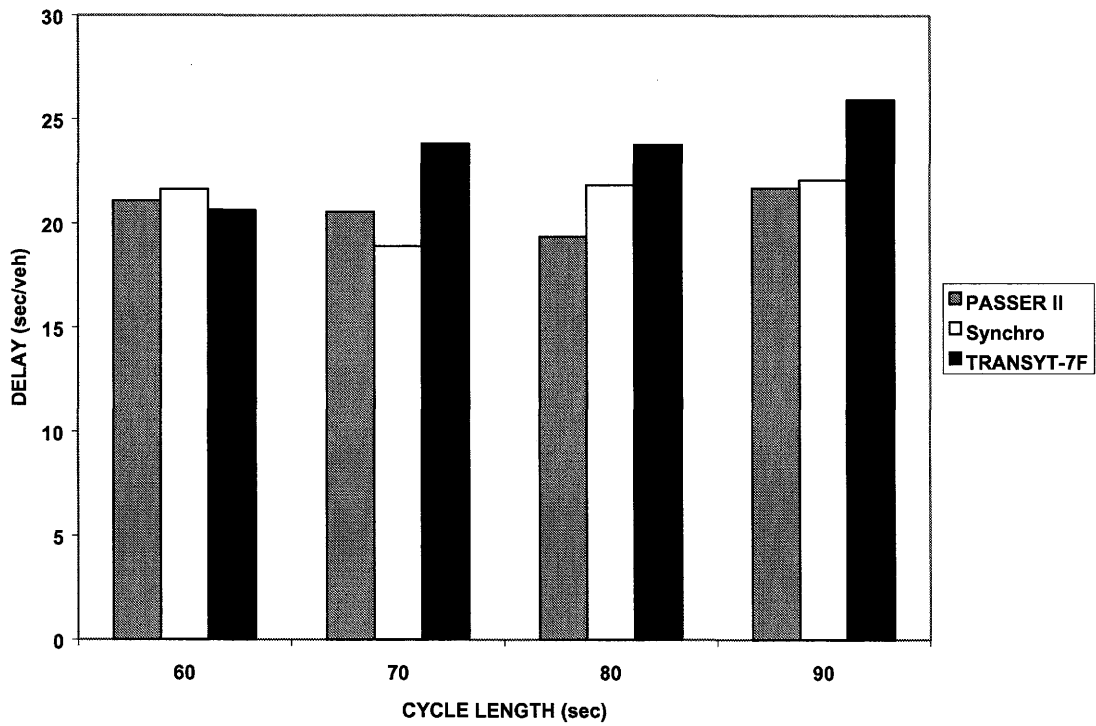
Figure B20. Comparison of Delay per Vehicle by Cycle Length for Artery 3 (GF=1.0).
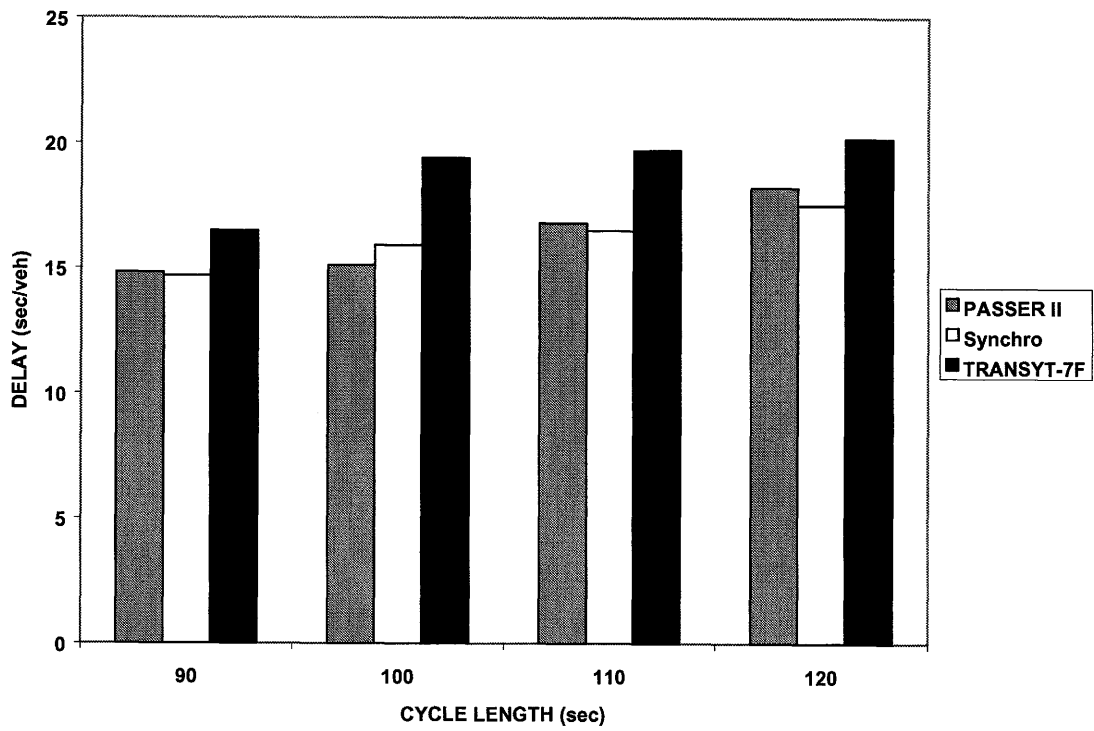
Figure B21. Comparison of Delay per Vehicle by Cycle Length for Artery 4 (GF=1.0).
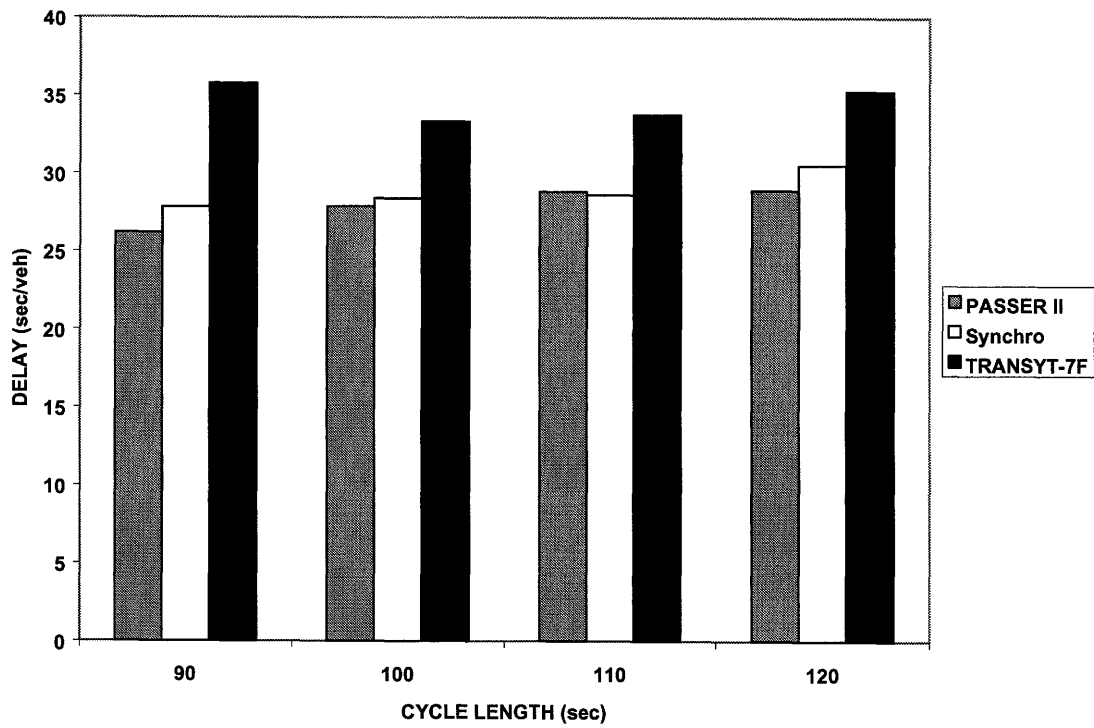
Figure B22. Comparison of Delay per Vehicle by Cycle Length for Artery 5 (GF=1.0).
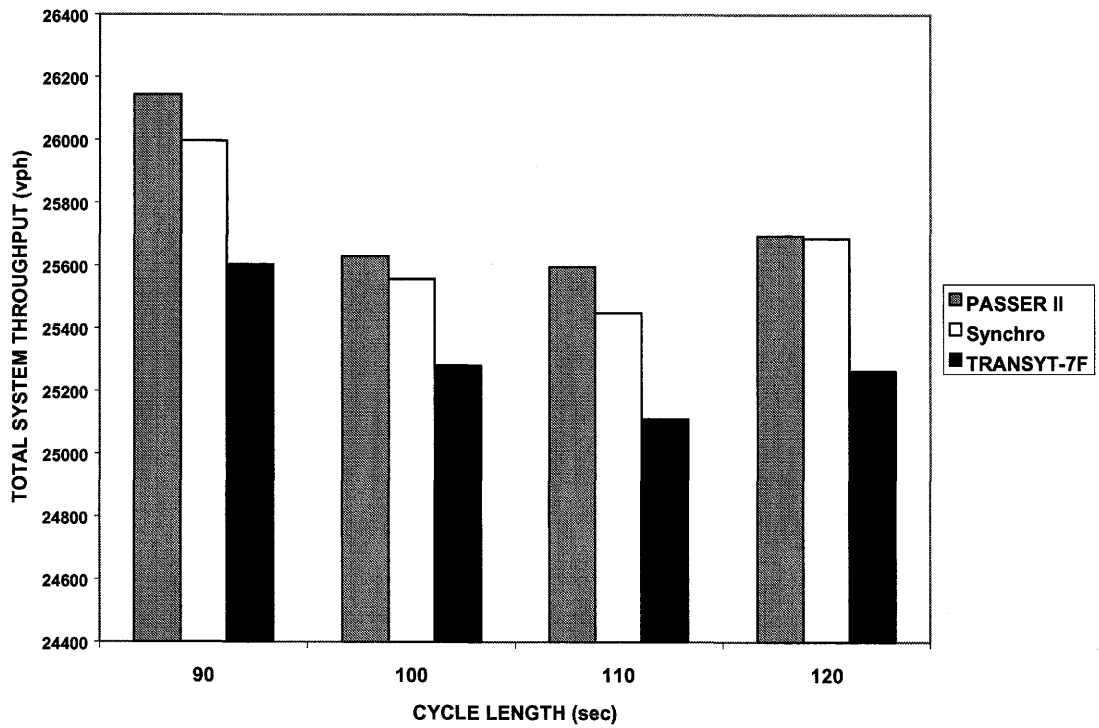
Figure B23. Comparison of System Throughput by Cycle Length for Artery 1 (GF=1.0).
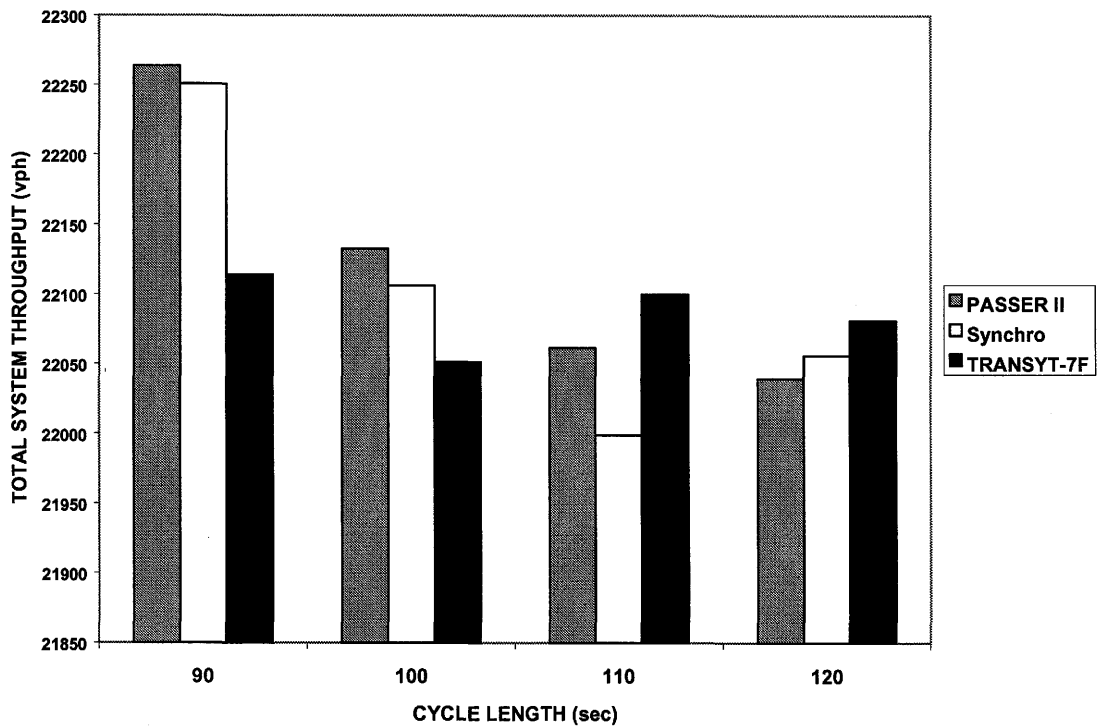
Figure B24. Comparison of System Throughput by Cycle Length for Artery 2 (GF=1.0).
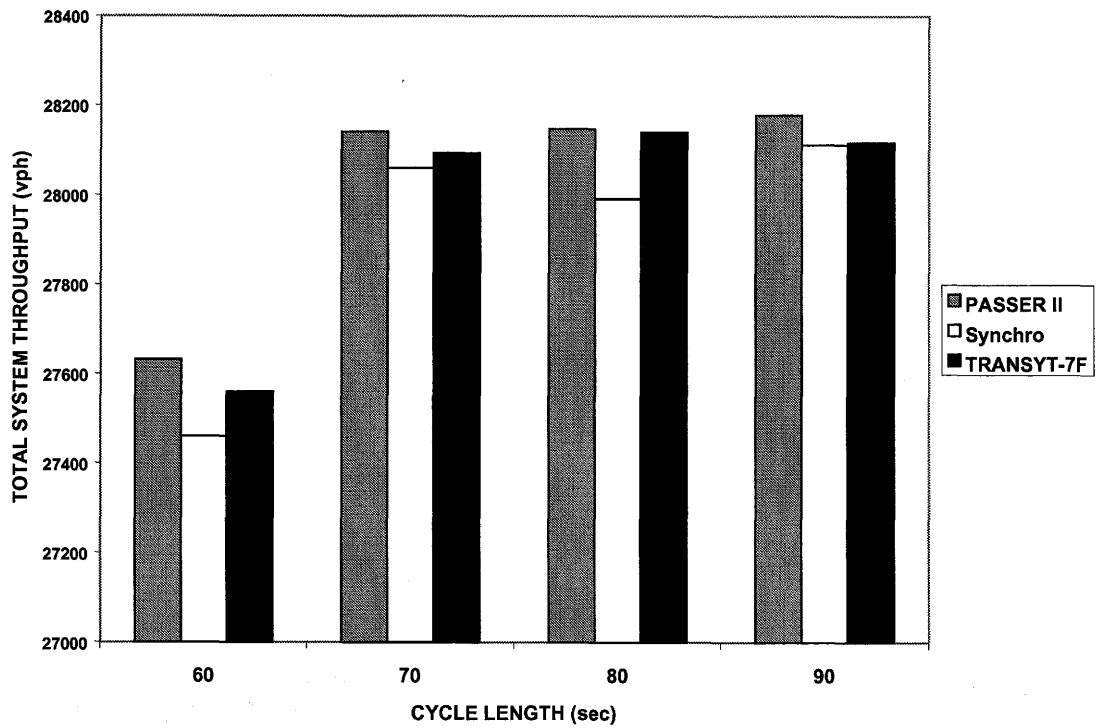
Figure B25. Comparison of System Throughput by Cycle Length for Artery 3 (GF=1.0).
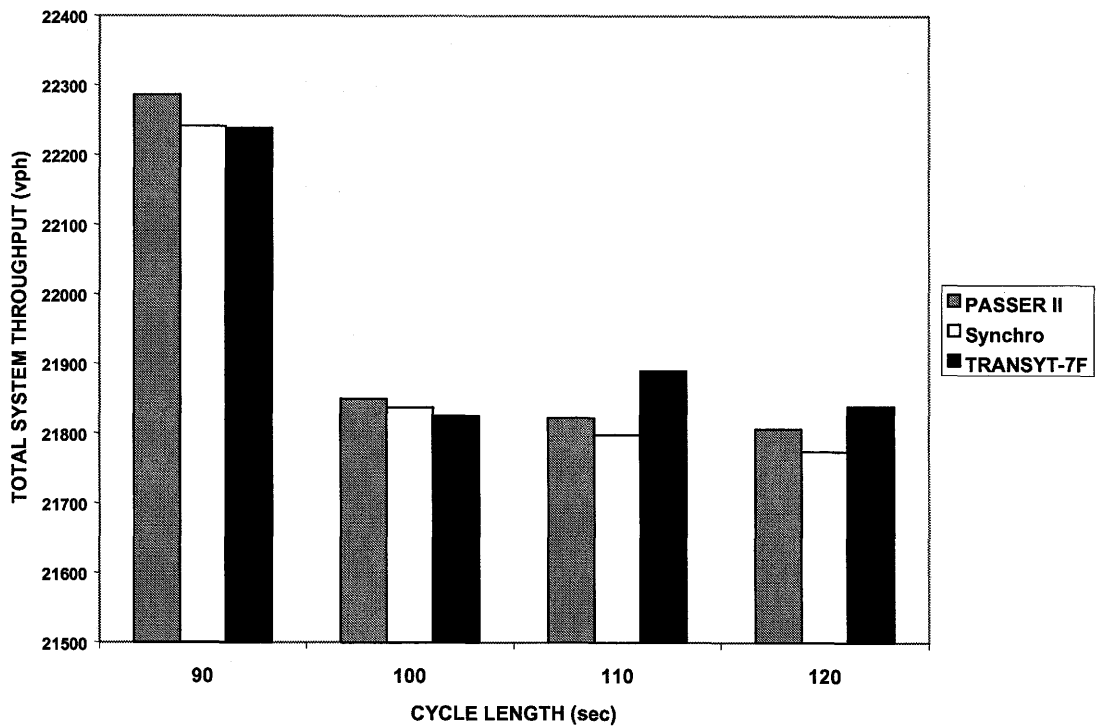
Figure B26. Comparison of System Throughput by Cycle Length for Artery 4 (GF=1.0).
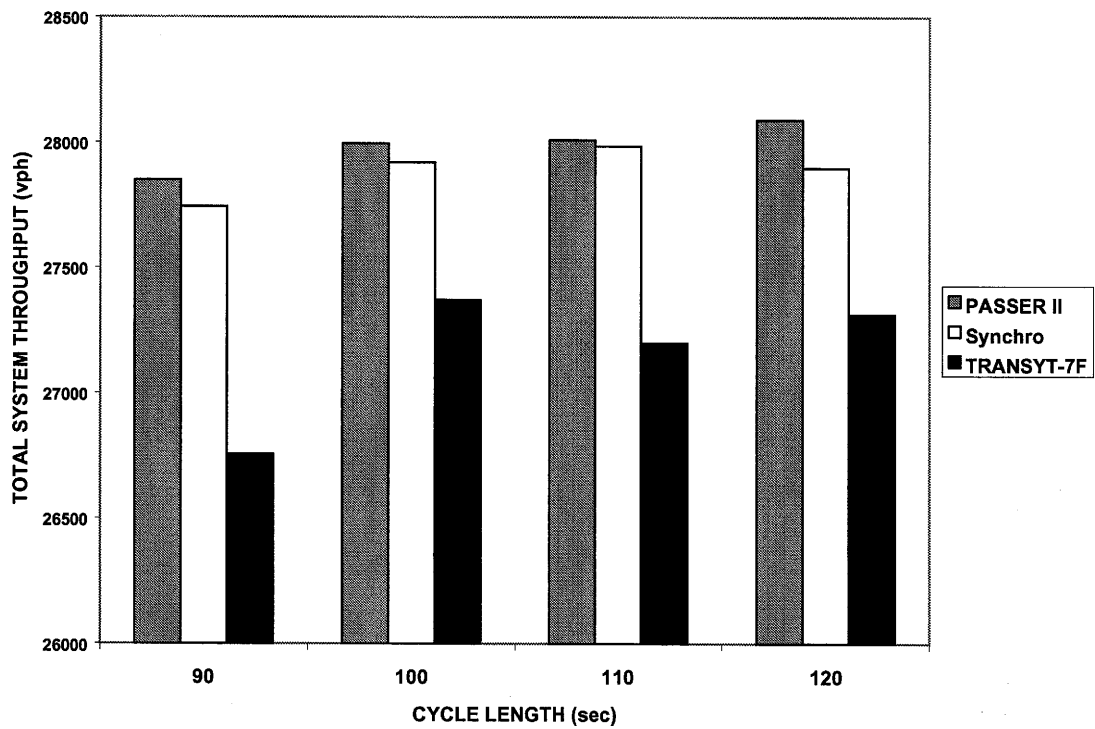
Figure B27. Comparison of System Throughput by Cycle Length for Artery 5 (GF=1.0).
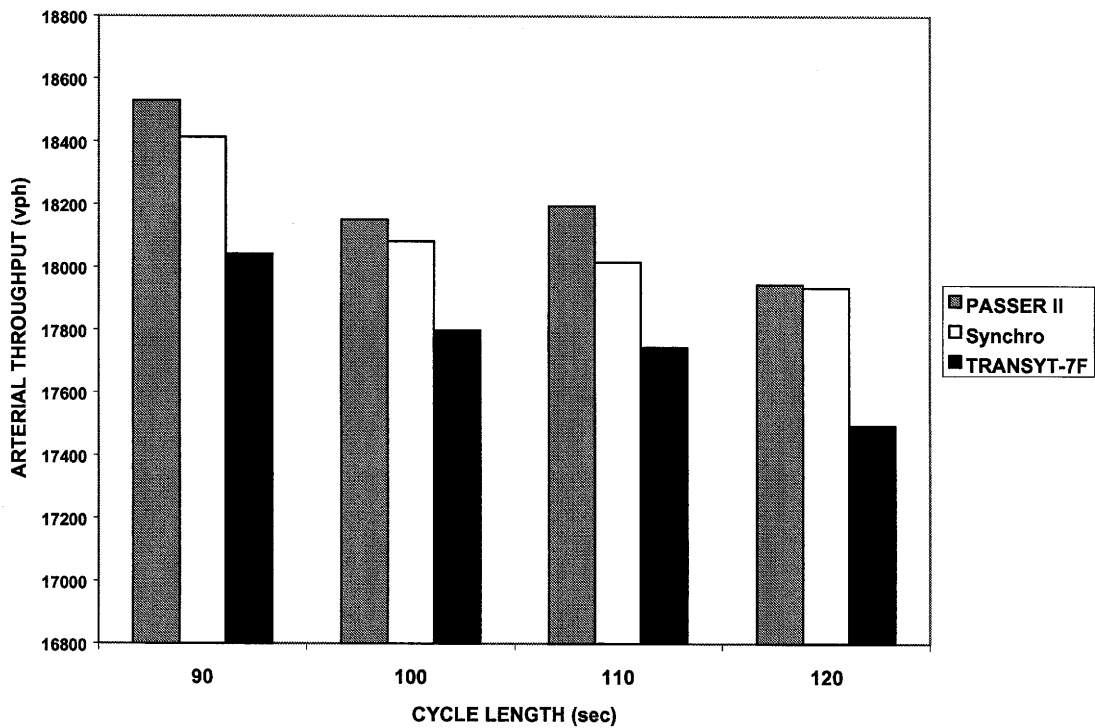
Figure B28. Comparison of Arterial Throughput by Cycle Length for Artery 1 (GF=1.0).
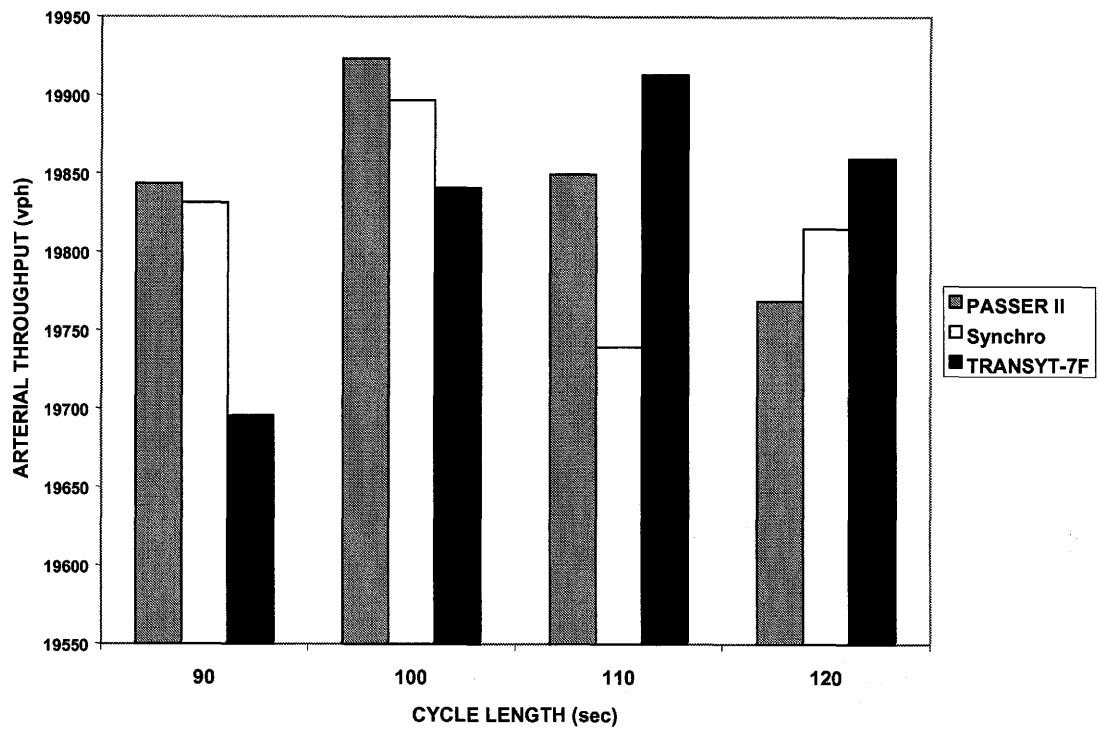
Figure B29. Comparison of Arterial Throughput by Cycle Length for Artery 2 (GF=1.0).
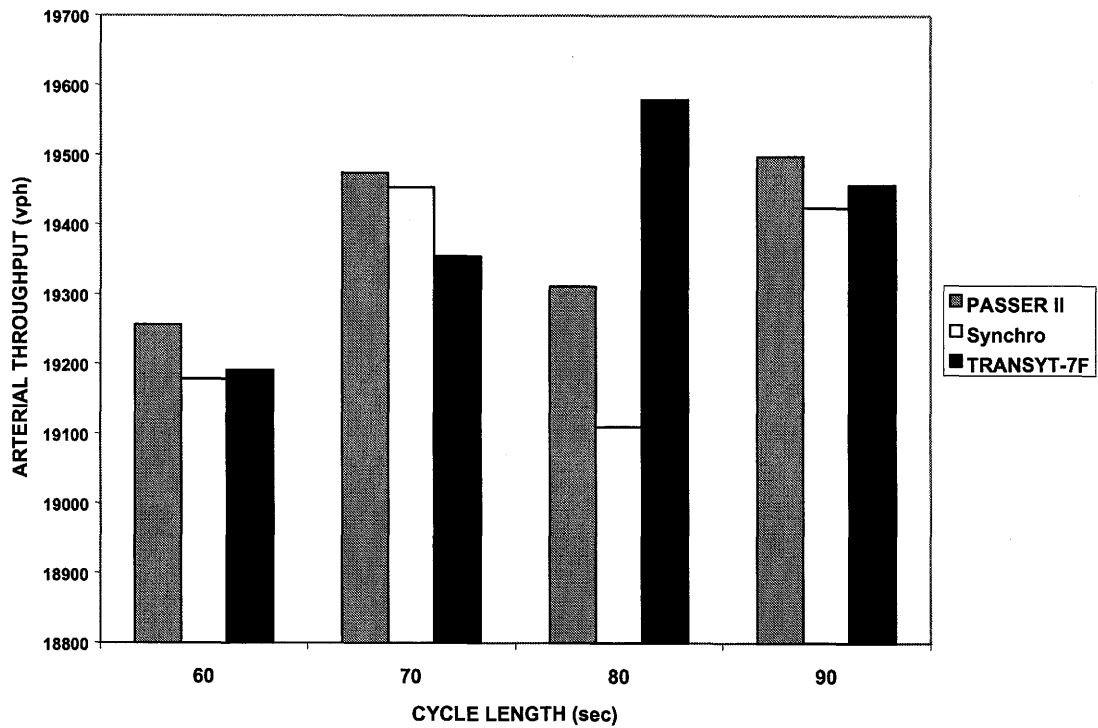
Figure B30. Comparison of Arterial Throughput by Cycle Length for Artery 3 (GF=1.0).
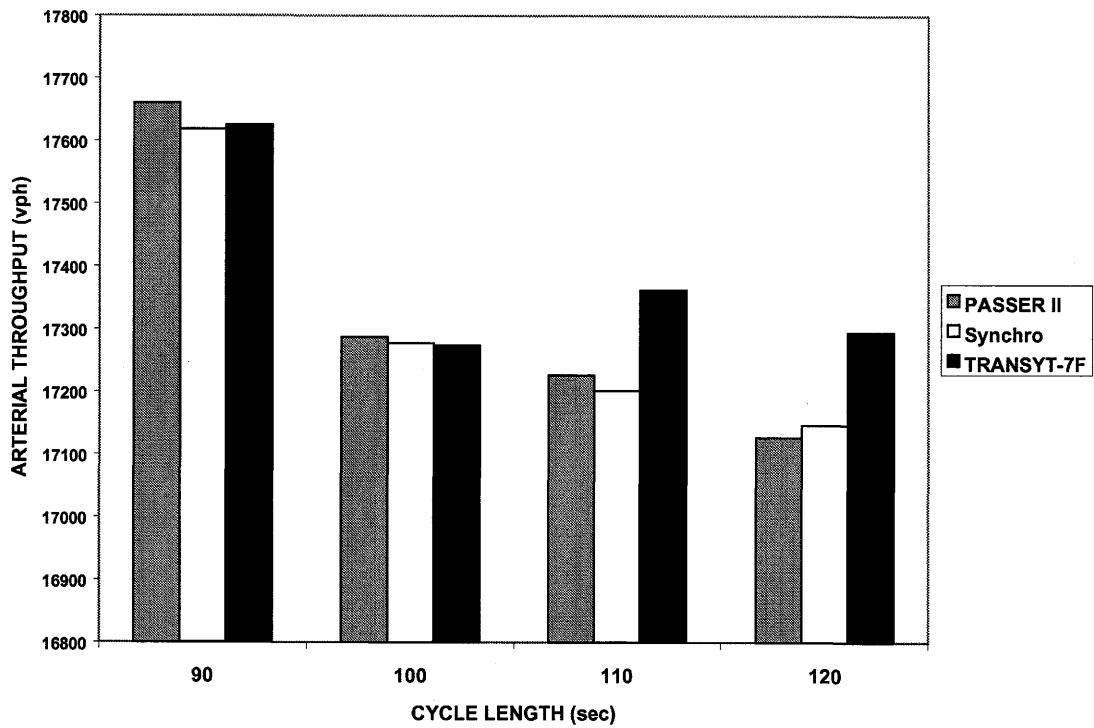
Figure B31. Comparison of Arterial Throughput by Cycle Length for Artery 4 (GF=1.0).
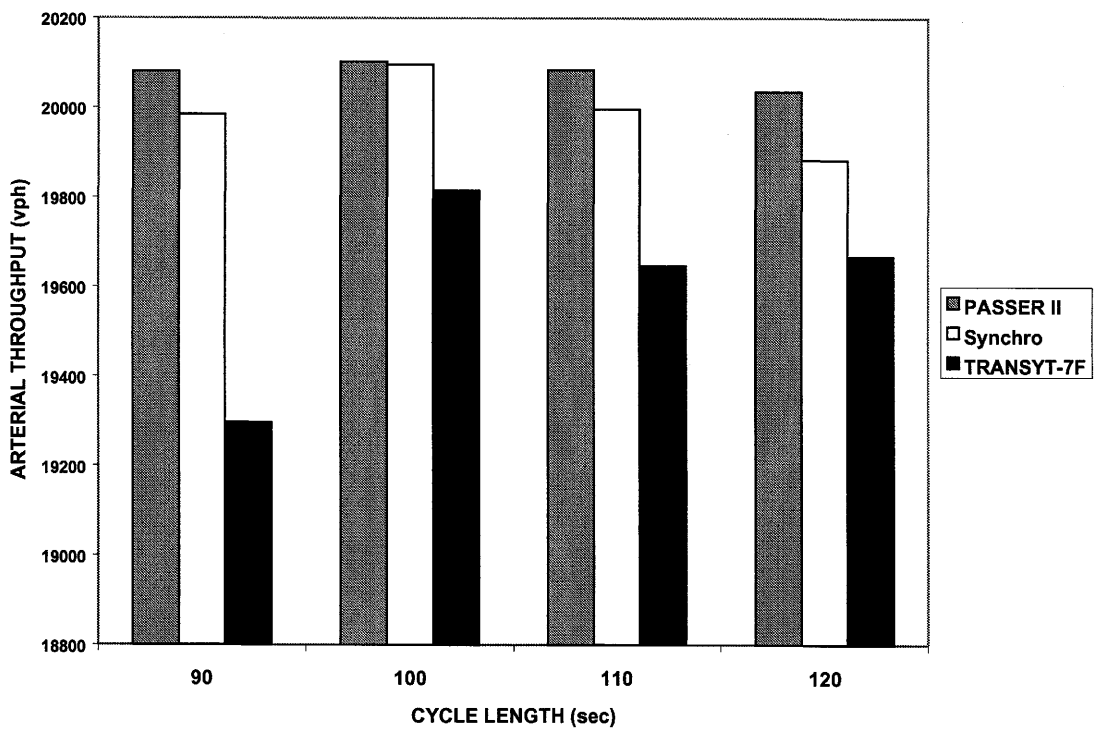
Figure B32. Comparison of Arterial Throughput by Cycle Length for Artery 5 (GF=1.0).

APPENDIX C. ANALYSIS RESULTS FOR LARGER ARTERIALS
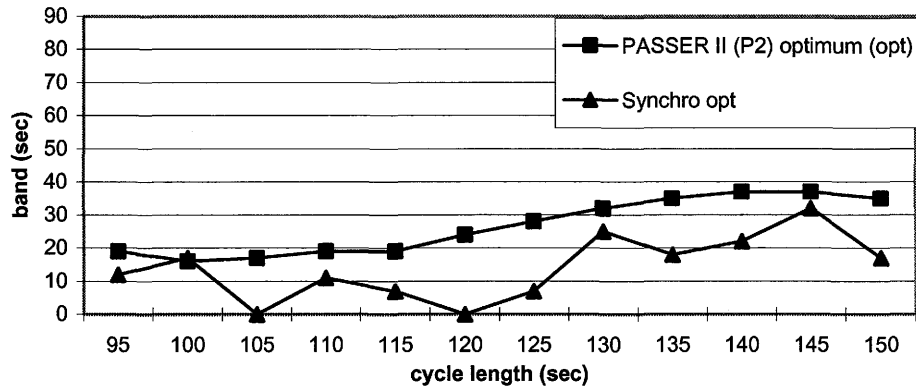
Figure C1. Bands from Synchro and PASSER II for Stevens Creek AM-Case.
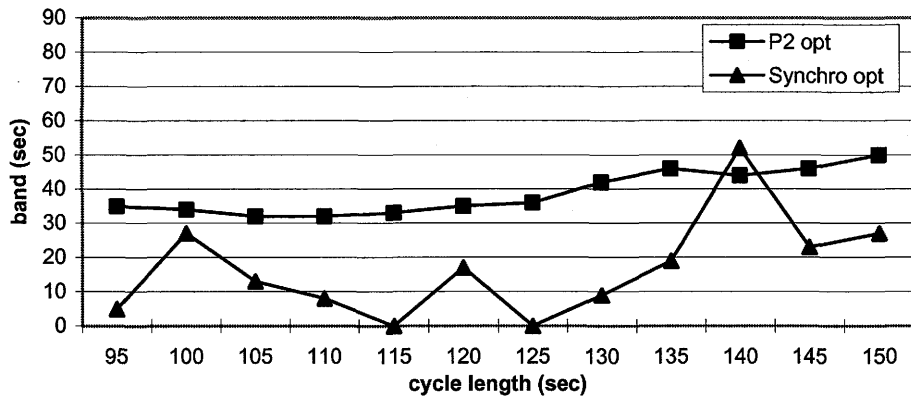
Figure C2. Bands from Synchro and PASSER II for Stevens Creek PM-Case.
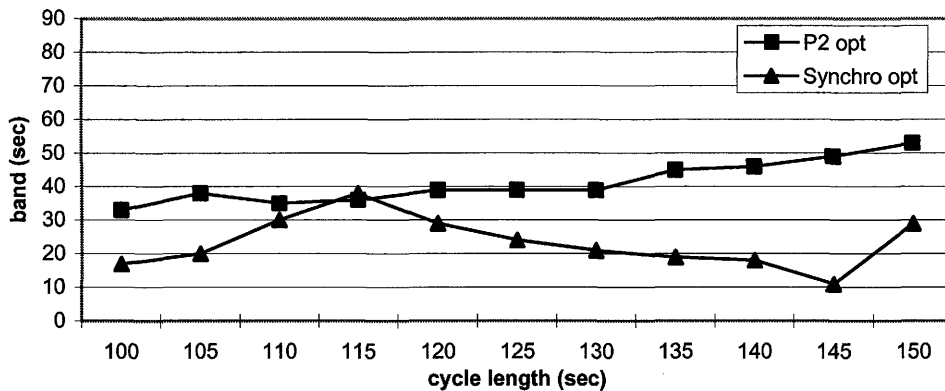
Figure C3. Total Bands from Synchro and PASSER II for De Anza Blvd. AM-Case.
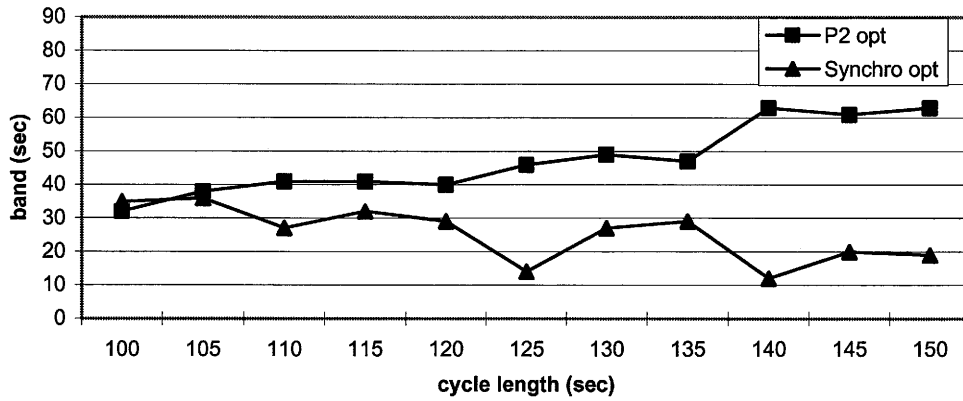
Figure C4. Total Bands from Synchro and PASSER II for De Anza Blvd. PM-Case.
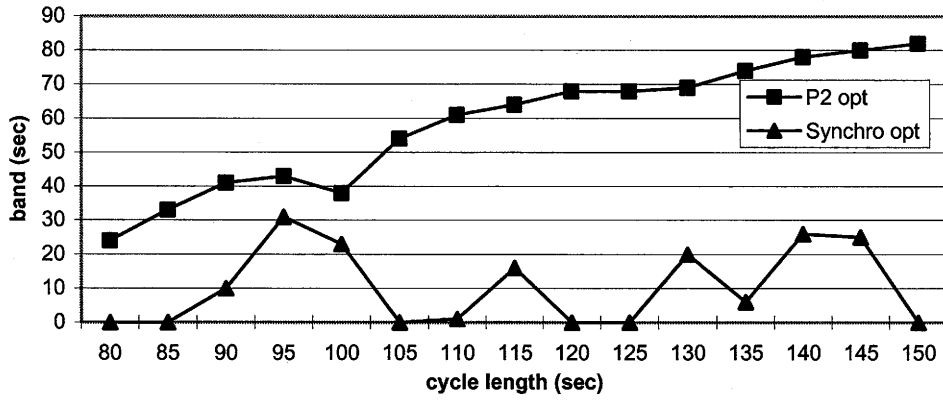
Figure C5. Total Bands from Synchro and PASSER II for US-90A.
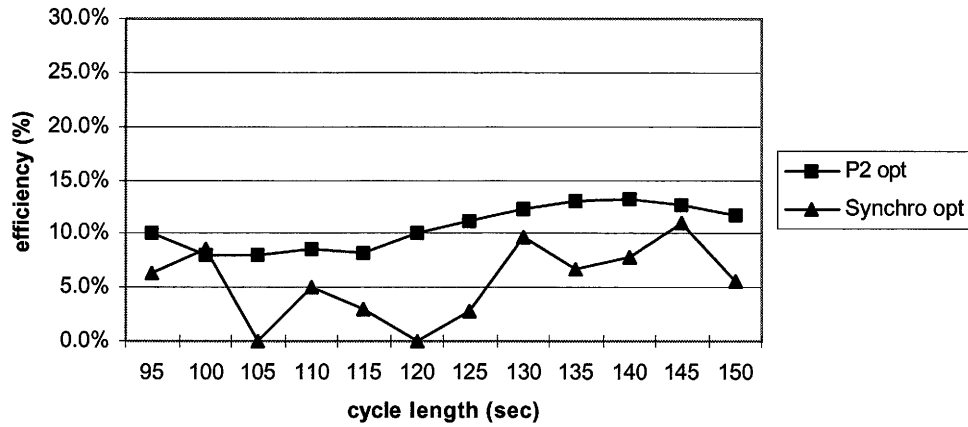
Figure C6. Band Efficiency from Synchro and PASSER II for Stevens Creek AM-Case.
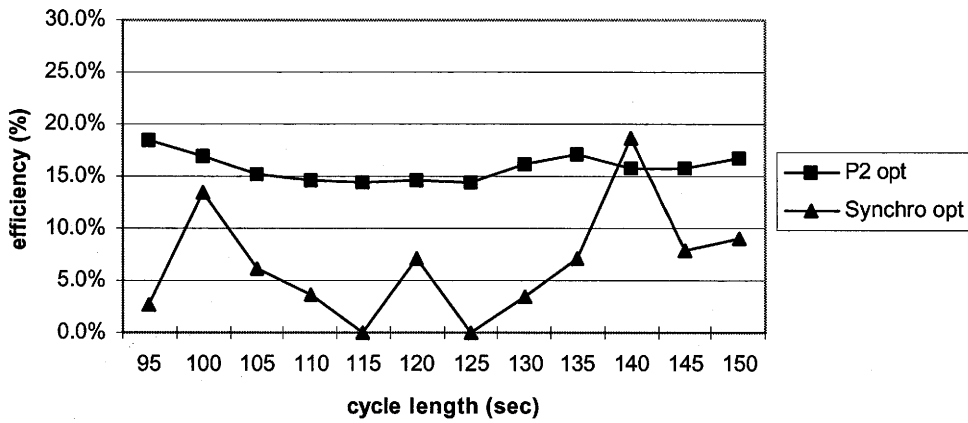
Figure C7. Band Efficiency from Synchro and PASSER II for Stevens Creek PM-Case.
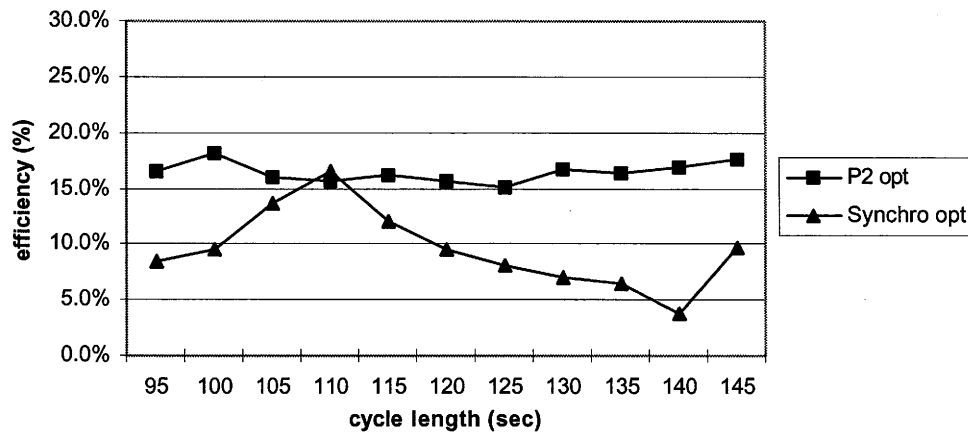
Figure C8. Band Efficiency from Synchro and PASSER II for De Anza Blvd. AM-Case.
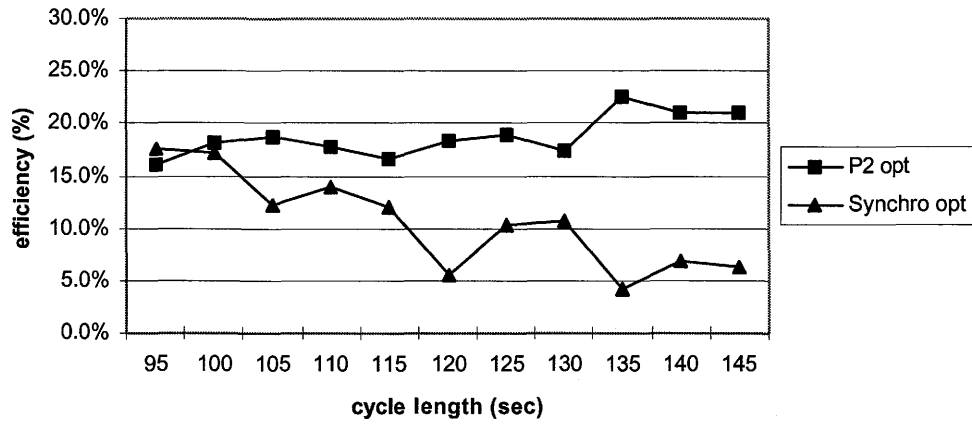
Figure C9. Band Efficiency from Synchro and PASSER II for De Anza Blvd. PM-Case.
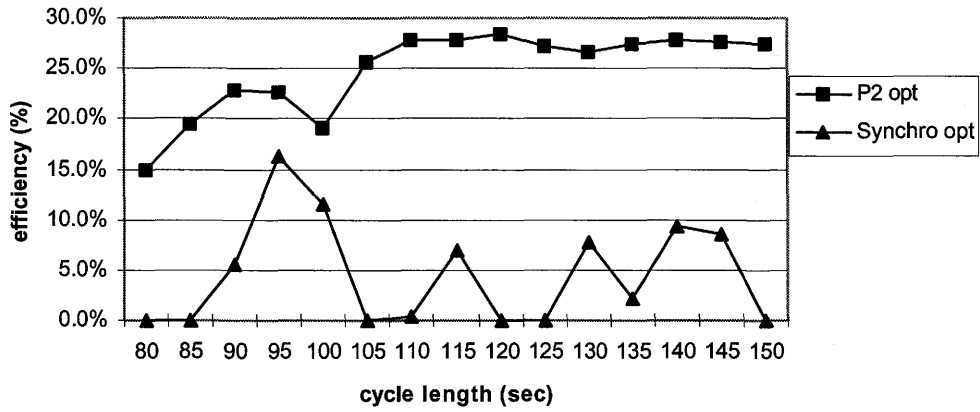
Figure C10. Band Efficiency from Synchro and PASSER II for US-90A.
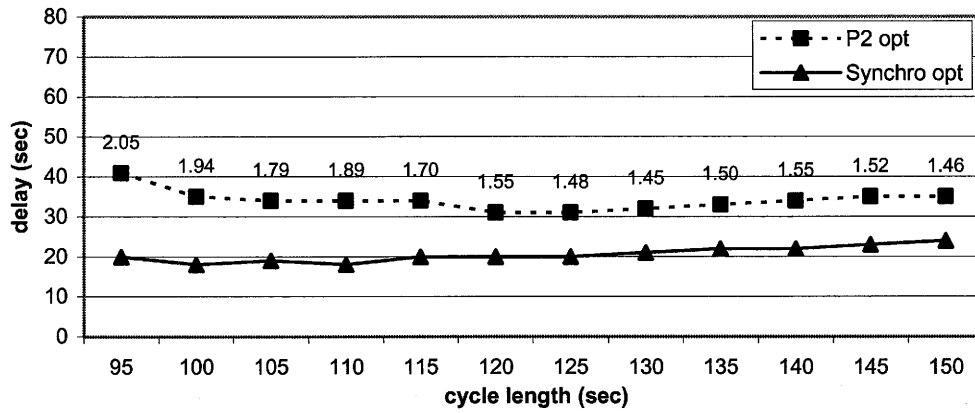
Figure C11. PASSER II and Synchro Delay for Stevens Creek AM-Case.
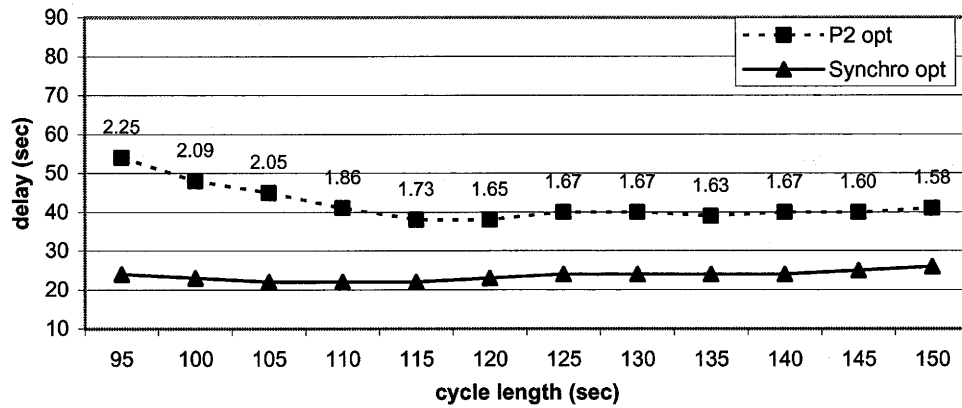
Figure C12. PASSER II and Synchro Delay for Stevens Creek PM-Case.
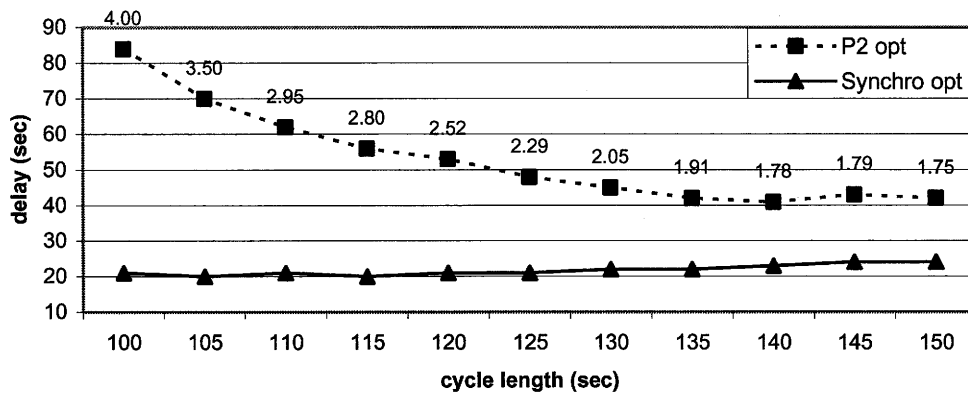
Figure C13. PASSER II and Synchro Delay for De Anza Blvd. AM-Case.

99
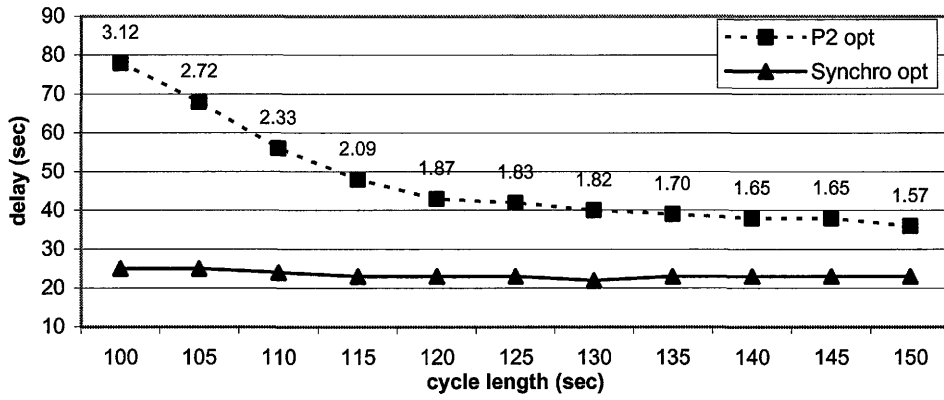
Figure C14. PASSER II and Synchro Delay for De Anza Blvd. PM-Case.
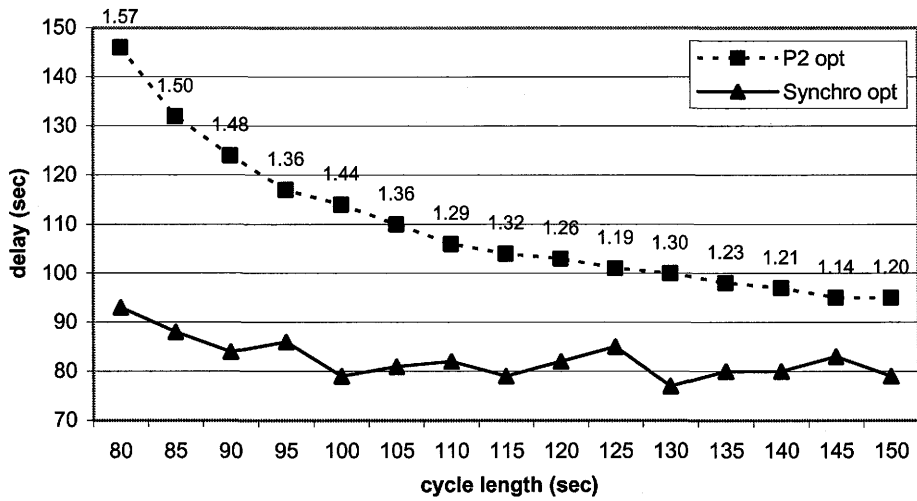
Figure C15. PASSER II and Synchro Delay for US-90A.
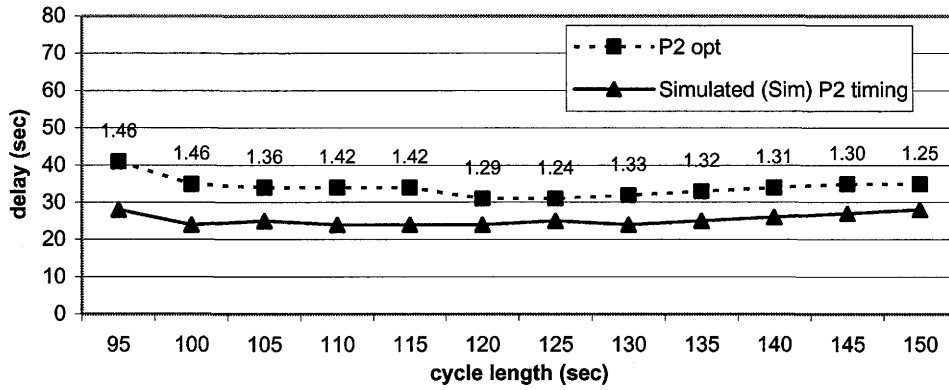
Figure C16. Simulation of PASSER II Timings Using Synchro for Stevens Creek AM-Case.
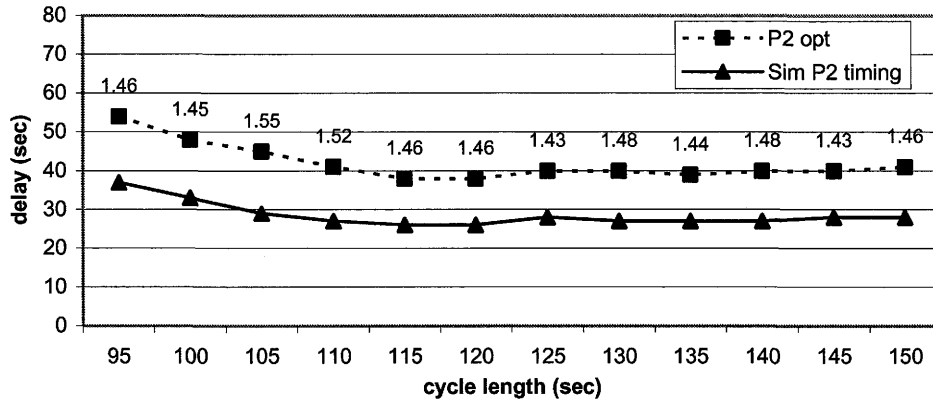
Figure C17. Simulation of PASSER II Timings Using Synchro for Stevens Creek PM-Case.
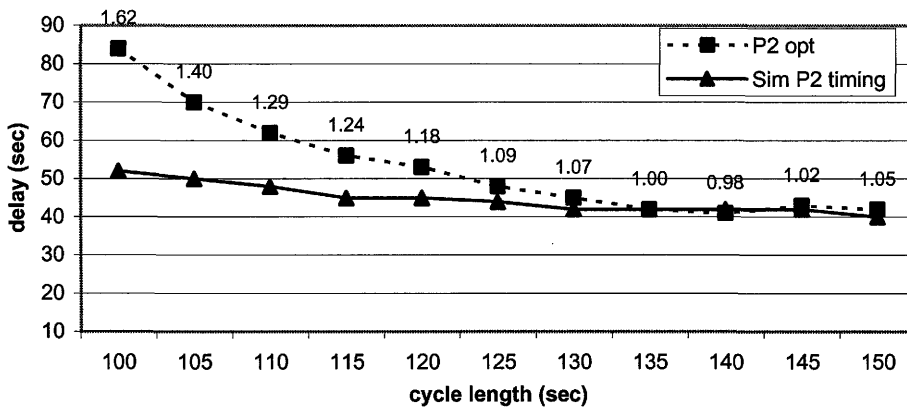
Figure C18. Simulation of PASSER II Timing Using Synchro for De Anza Blvd AM-Case.

101
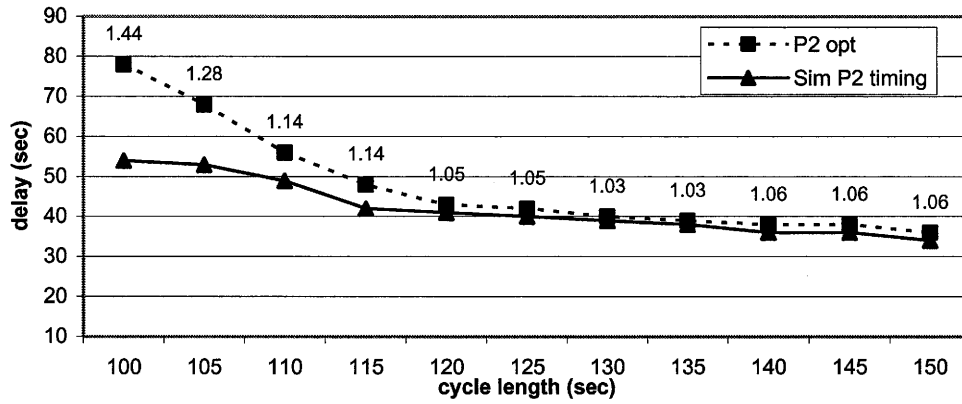
Figure C19. Simulation of PASSER II Timings Using Synchro for De Anza Blvd. PM-Case.
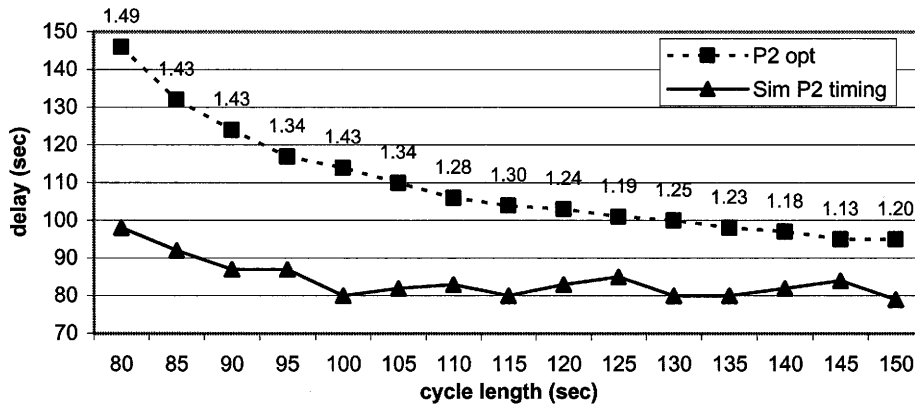
Figure C20. Simulation of PASSER II Timings Using Synchro for US-90A.

102
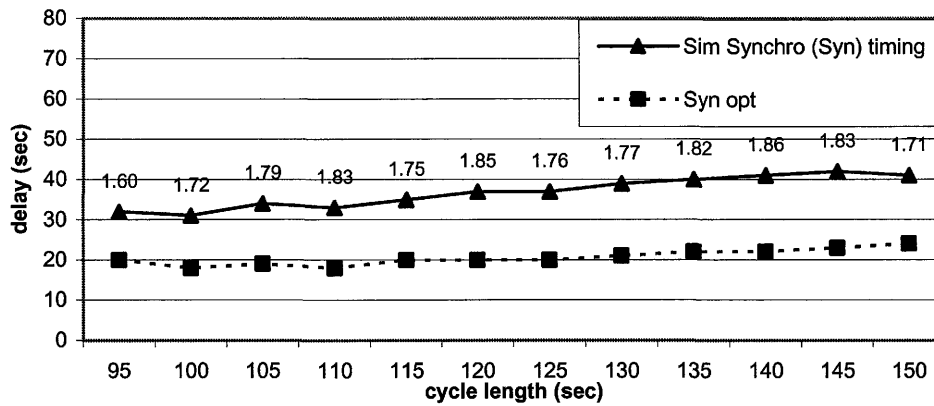
Figure C21. Simulation of Synchro Timings Using PASSER II for Stevens Creek AM-Case.
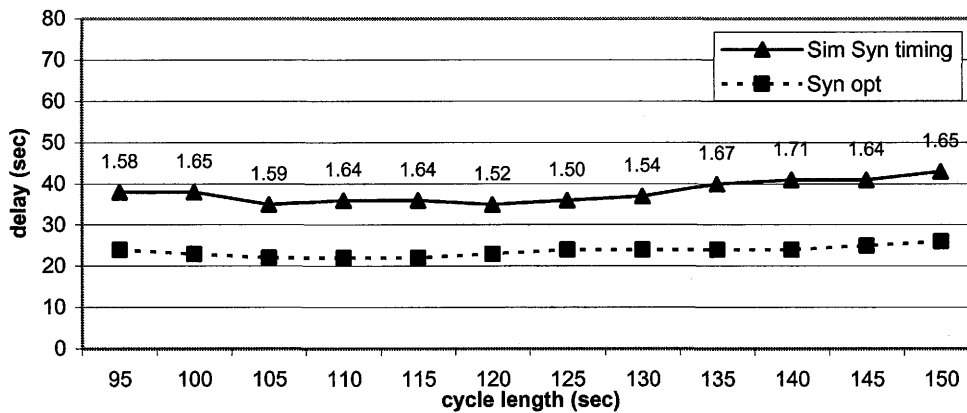
Figure C22. Simulation of Synchro Timings Using PASSER II for Stevens Creek PM-Case.
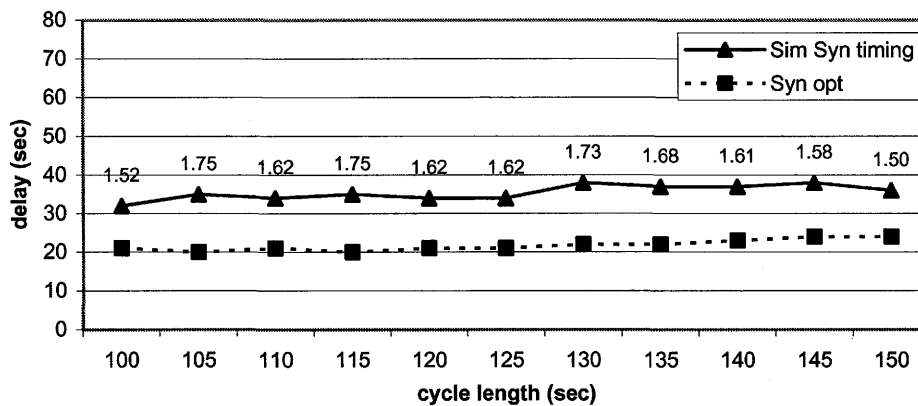
Figure C23. Simulation of Synchro Timings Using PASSER II for De Anza Blvd. AM-Case.

103
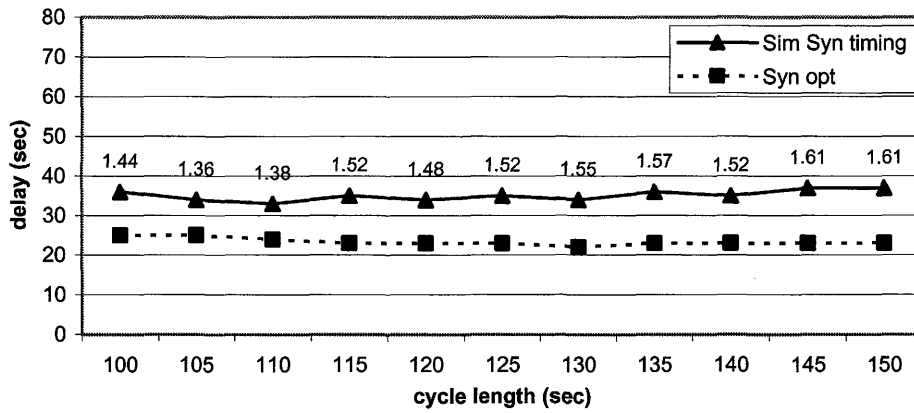
Figure C24. Simulation of Synchro Timings Using PASSER II for De Anza Blvd. PM-Case.
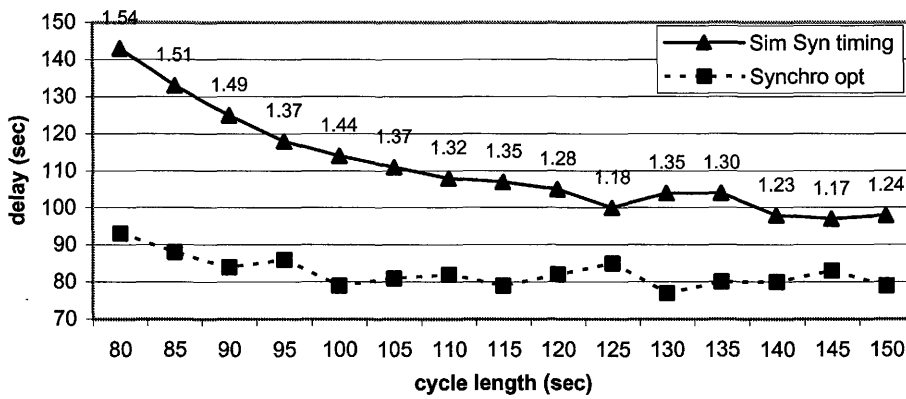
Figure C25. Simulation of Synchro Timings Using PASSER II for US-90A.
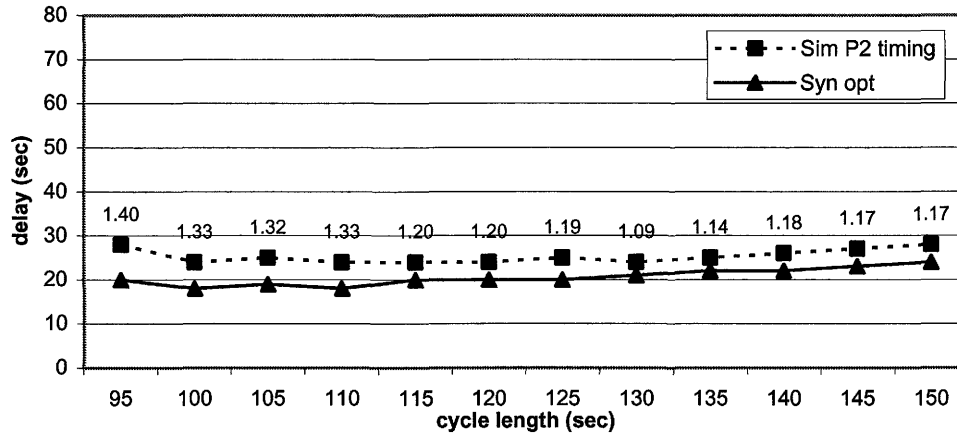
Figure C26. Synchro Assessment of Timings from Both Programs for Stevens Creek AM.
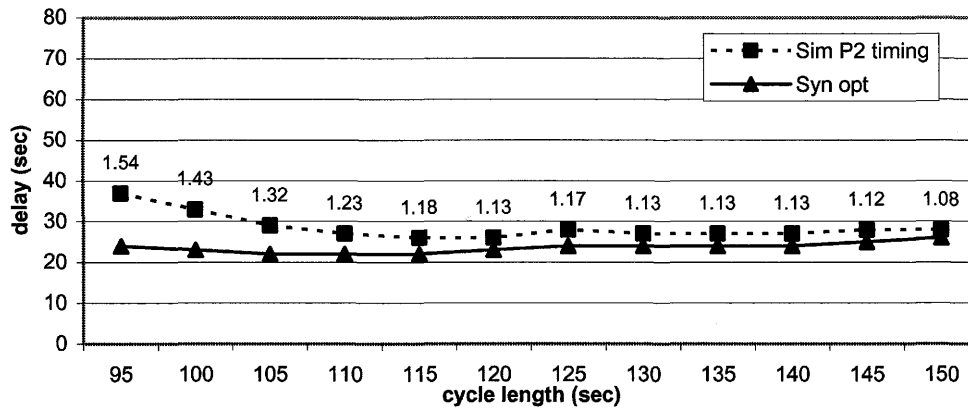
Figure C27. Synchro Assessment of Timings from Both Programs for Stevens Creek PM.
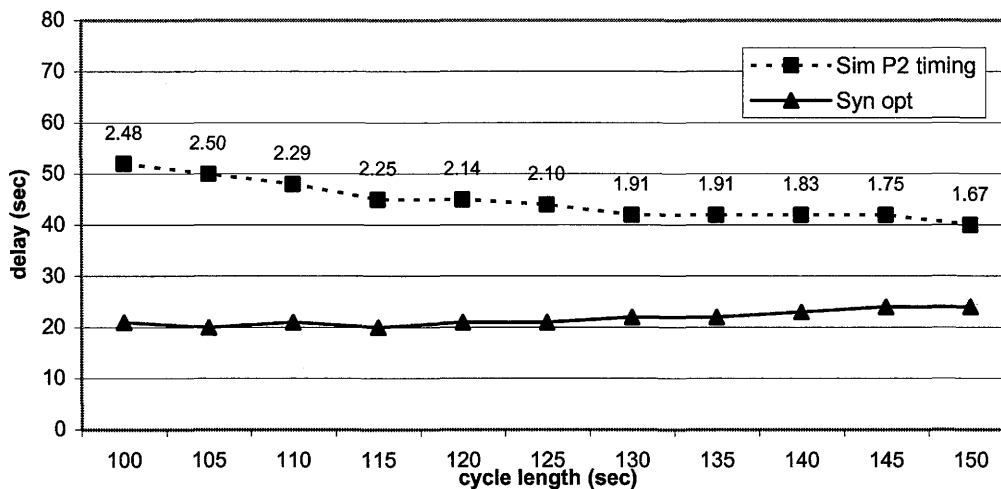
Figure C28. Synchro Assessment of Timings from Both Programs for De Anza Blvd. AM.

105
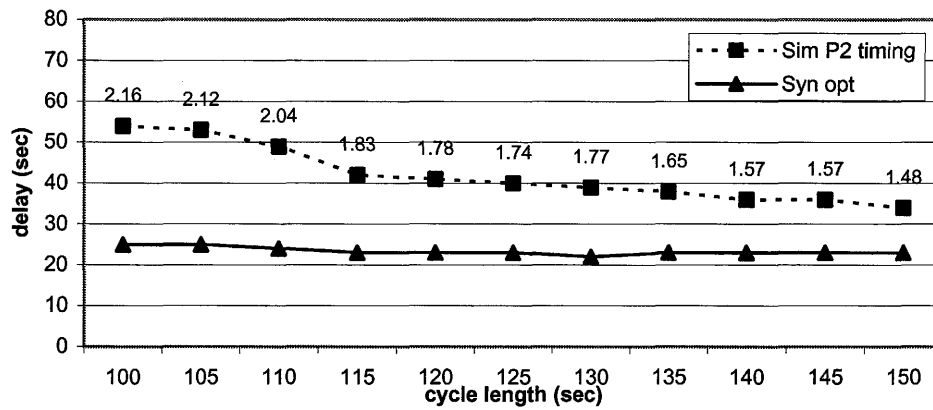
Figure C29. Synchro Assessment of Timings from Both Programs for De Anza Blvd. PM.
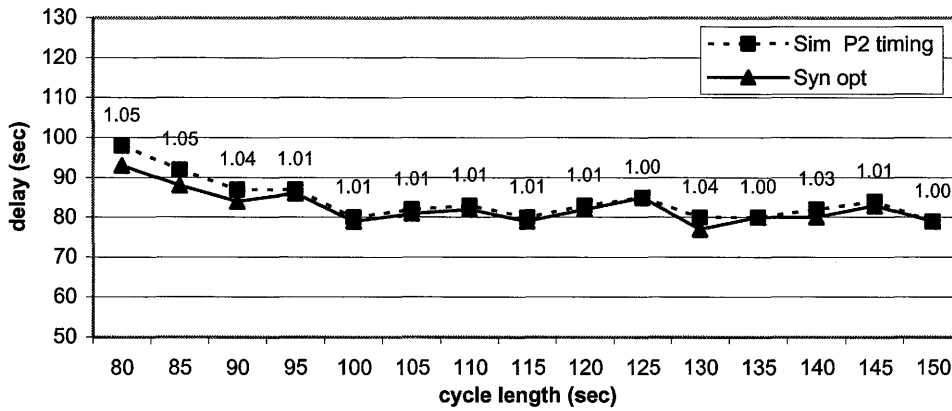
Figure C30. Synchro Assessment of Timings from Both Programs for US-90A.
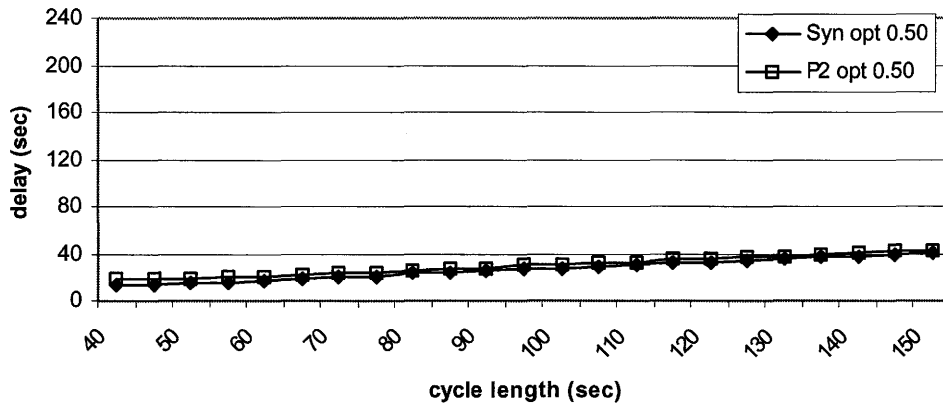
Figure C31. Synchro and PASSER II Delays for Isolated Signal with Volumes at GF 0.50.
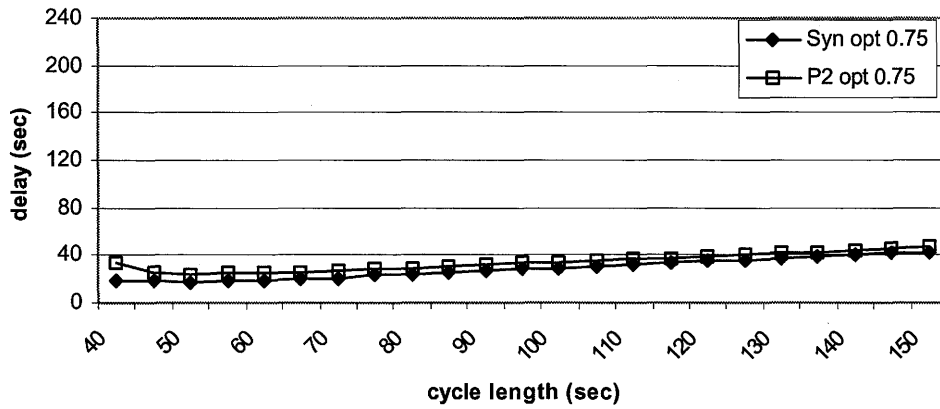
Figure C32. Synchro and PASSER II Delays for Isolated Signal with Volumes at GF 0.75.
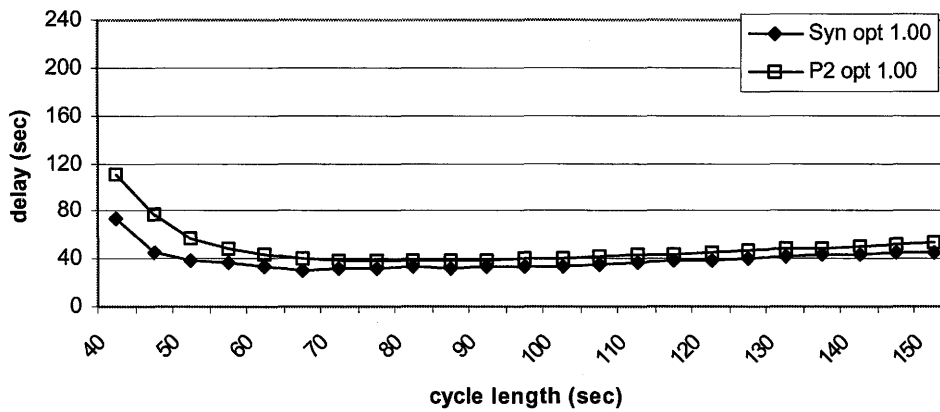
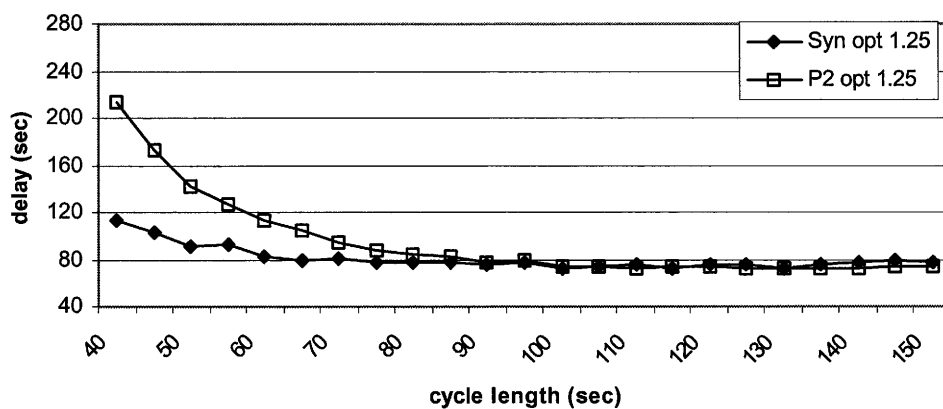**Figure C33. Synchro and PASSER II Delays for Isolated Signal with Volumes at GF 1.00.**

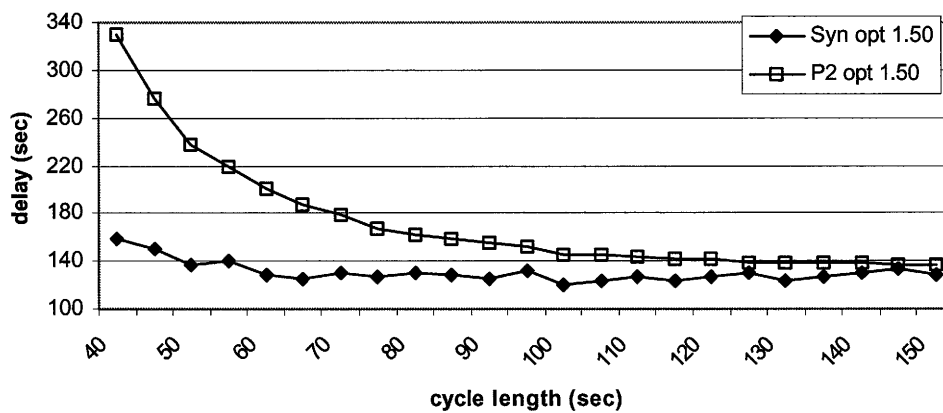**Figure C34. Synchro and PASSER II Delays for Isolated Signal with Volumes at GF 1.25.**

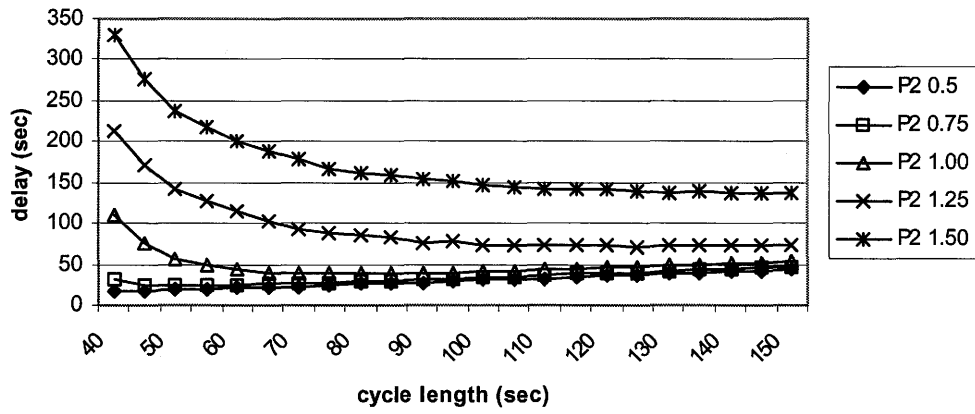**Figure C35. Synchro and PASSER II Delays for Isolated Signal with Volumes at GF 1.50.**

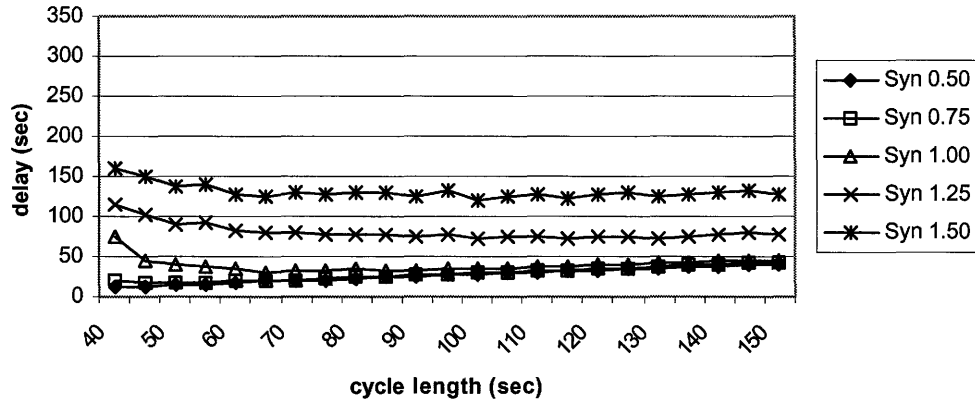**Figure C36. Delay from PASSER II for All Isolated-Signal Scenarios.**

Figure C37. Delay from Synchro for All Isolated-Signal Scenarios.

109